



Pexip Infinity

Administrator Guide

Software Version 26

Document Version 26.a

July 2021

] pexip [

Contents

The Pexip Infinity platform	9
Introduction to Pexip Infinity	10
Scheduling and joining meetings	10
Management and control	11
Privacy and security	11
Customizable to your standards	11
Pexip Infinity features and specifications	12
Pexip Infinity platform	12
Pexip Infinity Connect	15
Audio and video specifications and codecs	16
Host hardware requirements	17
Capacity	18
Hypervisor requirements	18
What's new in version 26?	19
Pexip Infinity new features and enhancements	19
Pexip Infinity changes in functionality	21
Planned changes in future releases	23
Infinity Connect web app new features and changes	23
Components of the Pexip Infinity platform	24
Management Node	24
Conferencing Nodes	25
Pexip Infinity Connect clients	25
Pexip Infinity APIs and SDKs	26
Conference types and services	26
VMR self-service portal	26
Host servers	26
Hypervisors	27
Call control	27
Distributed architecture	29
Benefits of the Pexip Infinity distributed architecture	29
Conference distribution	30
Distributed Proxying Edge Nodes	32
Bandwidth optimization	33
Load balancing, redundancy and scalability	35
Customizing the Pexip Infinity user experience	37
Themes	37
Infinity Connect customization	38
A beginner's guide to Pexip Infinity	40
First, a bit of history	40
What's wrong with using what I have now?	40
What to look for in a modern videoconferencing solution	40
The advantages of software	41
Next steps	41
Using your Virtual Meeting Room	42

It's always available	42
VMR addresses and PINs	42
Use any device	42
Using your keypad to control the conference	42
Try out the features in Infinity Connect!	42
Connecting with Skype for Business / Lync	43
Pexip Infinity installation guidelines	44
Installation overview	45
Planning and prerequisites	45
Choosing a deployment environment	45
Server and network requirements	46
Capacity planning	47
Supported hypervisors	50
About the Pexip Infinity software files	51
Network deployment options	52
Deployment guidelines for Proxying Edge Nodes	59
Network routing and addressing options for Conferencing Nodes	67
Firewall/NAT routing and addressing examples	70
Dynamic bursting to a cloud service	73
Handling of media and signaling	76
Implementing a dial plan	84
DNS record examples	86
Testing and next steps after initial installation	90
Making a test call	90
Further configuration	90
Integrating with a call control system	90
Configuring the Pexip Infinity Distributed Gateway	91
Registering devices directly to the Pexip Infinity platform	91
Customizing the user experience	91
Informing users about the new video conferencing service	91
Administering Pexip Infinity	92
Using the Pexip Infinity Administrator interface	93
Accessing the Pexip Infinity Administrator interface	93
Setting the session timeout	93
Changing the display language	93
Timezones	94
Getting help and support	94
Pexip Infinity system configuration	95
Configuring DNS servers	95
Syncing with NTP servers	95
Using a web proxy	96
Monitoring via SNMP	96
Using a syslog server	99
Configuring SMTP servers	101
Managing static routes	101
Using event sinks to monitor conference and participant status	102

Pexip Infinity platform configuration	105
Configuring the Management Node	105
About global settings	107
About system locations	115
Enabling Pexip Smart Scale	121
About H.323 gatekeepers and SIP proxies	127
About Skype for Business / Lync servers	128
Using TURN servers with Pexip Infinity	129
Using STUN servers with Pexip Infinity	131
Configuring policy profiles	132
Pexip Infinity license installation and usage	134
Managing TLS and trusted CA certificates	139
Certificate signing requests (CSRs)	145
Verifying SIP TLS connections with peer systems	148
Integrating with external systems	150
Enabling and disabling SIP, H.323, WebRTC and RTMP	151
Break-in resistance settings to mitigate rogue calls	151
Enabling and disabling chat messages	154
Conferencing Node configuration	156
Deploying new Conferencing Nodes	156
Deploying a Conferencing Node on an ESXi host	157
Deploying a Conferencing Node	160
Deploying a Conferencing Node on a KVM host	163
Deploying a Conferencing Node on a Xen host	168
Deploying a Conferencing Node using a generic VM template and configuration file	172
Assigning hostnames and FQDNs	175
Configuring existing Conferencing Nodes	175
Deleting Conferencing Nodes	178
Pexip Infinity conference types	180
About Pexip Infinity conferences	180
About Virtual Meeting Rooms (VMRs) and Virtual Auditoriums	180
Configuring Virtual Meeting Rooms (VMRs)	181
Configuring Virtual Auditoriums	185
About the Virtual Reception IVR service	188
Configuring Virtual Reception IVRs	190
Placing calls via the Pexip Infinity Distributed Gateway	193
Configuring Call Routing Rules	196
Configuring the Test Call Service	204
Registering devices to Pexip Infinity	207
Registering and provisioning the Infinity Connect desktop client	215
Customizing the Infinity Connect clients	222
Pexip Infinity conference settings	228
About aliases and access numbers	228
About PINs, Hosts and Guests	231
Selecting the layout seen by participants	235
Limiting the number of participants	245
Automatically dialing out to a participant from a conference	245
Automatically ending a conference	249
Controlling media capability	250

Streaming and recording a conference	251
Setting and limiting call quality	254
Managing and restricting call bandwidth	256
Enabling and disabling chat messages	258
Playing notification tones when participants join or leave a conference	259
Controlling active conferences	260
Locking a conference and allowing participants to join a locked conference	260
Muting a participant's audio	261
Manually dialing out to a participant from a conference	263
Disconnecting participants from a conference	266
Transferring a participant to another conference	267
Using a DTMF keypad to control a conference	268
Customizing with themes	269
Customizing conference images and voice prompts using themes	269
Creating and applying themes to conferences	271
Base theme and other preconfigured themes	274
Rules and requirements for customized themes	274
Integrating Google Meet with Pexip Infinity	299
Introduction	299
Configuring Google Workspace for Google Meet integration	302
Configuring Pexip Infinity as a Google Meet gateway	305
Integrating Microsoft Teams with Pexip Infinity	318
Integrating Epic telehealth with Pexip Infinity	319
Epic telehealth integration with Pexip Infinity	319
Configuring Pexip Infinity to integrate with Epic telehealth	321
Optional features and customizations for Epic telehealth integrations	330
Monitoring, maintenance and reference information for Epic telehealth integrations	333
Troubleshooting and call setup information for Epic telehealth integrations	336
Integrating Pexip Infinity with authentication and provisioning services	339
Managing administrator access via LDAP	339
Provisioning VMRs, devices and users from Active Directory via LDAP	349
Applying user records	361
Using templates, variables and filters when provisioning VMRs, devices and users	363
Sending provisioning emails to VMR and device owners	369
Troubleshooting LDAP server connections	375
Using AD FS for client authentication	378
Pexip Infinity maintenance tasks	387
Backing up and restoring configuration	387
Upgrading the Pexip Infinity platform	390
Setting and changing usernames and passwords	393
Re-running the installation wizard	395
Migrating Conferencing Nodes between host VMware servers	396
Taking a Conferencing Node out of service	396
Rebooting and shutting down a Conferencing Node	397
Moving, restoring or changing the IP address of the Management Node	398
Bulk import/export of service configuration data	402
Best practices	413

Performing routine checks	413
Security best practices	414
Resilience strategies — redundancy, backing up and restoring data	415
PSTN gateways and toll fraud	419
Example emails for sending to new users	420
Pexip Infinity reference information	422
Glossary of Pexip Infinity terms	423
Regular expression (regex) reference	428
Regex testing tool	428
Regex syntax	428
Pattern matching examples	430
Search and replace examples	430
Jinja2 templates and filters	432
Template content	432
Supported jinja2 filters	432
Custom Pexip filters	433
Encryption methodologies	435
Pexip nodes	435
Endpoints	435
Interoperability	436
Supported RFCs	437
Patents	439
Accessibility	440
Using Microsoft Skype for Business / Lync with Pexip Infinity	441
Architecture options	441
When is a reverse proxy, TURN server or STUN server required?	443
Integrating with streaming and recording services	444
Streaming a conference to YouTube	444
Streaming a conference to Facebook	452
Streaming a conference to Periscope	458
Streaming a conference to Wowza Streaming Cloud	460
Streaming a conference to Microsoft Stream	462
Integrating with telephone systems (PSTN)	467
Overview	467
Prerequisites	468
Example using Twilio	468
Pexip Infinity diagnostics	472
Viewing live and historical platform status	473
Key to icons and symbols	473
Platform summary status, call quality issues and alarms	475
Pie charts and detailed participant usage graphs	476
Viewing location status	477
Viewing Conferencing Node status	477
Viewing Teams Connector and Teams meeting status	477

Viewing conference status	478
Filtering by participant or conference	478
Rewinding and replaying status	479
Conference status	480
Viewing current conference status	480
Viewing participant status	487
Viewing registrations	491
Viewing historical information about conferences	491
Viewing historical information about participants	496
Reporting of media statistics and perceived call quality	499
Viewing usage statistics	499
Viewing LDAP sync template results	500
Diagnostics tools and reporting	503
Automatically reporting errors	503
Tracking usage via service and participant call tags	503
Automatically sending usage statistics	504
Downloading a diagnostic snapshot	511
Performing a network packet capture	511
Viewing Conferencing Nodes	514
Viewing current Conferencing Nodes	514
Viewing historic Conferencing Node events	515
Viewing system location status	516
Viewing cloud bursting status	517
Viewing current status	517
Viewing historic events	517
Viewing alarms	518
Cloud bursting alarms	524
One-Touch Join alarms	525
Viewing login history	528
About the support log	529
Viewing the support log	529
Searching the support log	530
Summarizing support log messages	530
File size	530
About the administrator log	531
Viewing the administrator log	531
Searching the administrator log	532
File size	532
Log output	533
Administrator log system modules	533
Support log system modules	543
Creating and viewing diagnostic graphs	546
Information shown in the graphs	546
Default graphs	546
Viewing and controlling graphs	546
Creating new graphs	547

Editing and deleting graphs	547
Disconnection reasons	548
Pexip Infinity port usage and firewall guidance	552
Firewall, routing and NAT guidance	552
Inter-node communication (Conferencing Nodes and Management Node)	552
Administration access	553
Peripheral services	553
Conferencing Node call signaling and media	556
Troubleshooting the Pexip Infinity platform	560
Pexip Infinity deployment and upgrading	560
Dynamic bursting to a cloud service	561
Joining a conference and viewing content	563
Conference connectivity and TLS issues	566
TLS certificate administration	566
Pexip Infinity administration	566
Infinity Connect clients	567

The Pexip Infinity platform

Pexip Infinity is a scalable meeting platform that connects virtually any communications tool, such as Microsoft Skype for Business / Lync, and traditional video and audio conferencing together for a seamless meeting experience.

In this section:

Introduction to Pexip Infinity	10
Pexip Infinity features and specifications	12
What's new in version 26?	19
Components of the Pexip Infinity platform	24
Distributed architecture	29
Customizing the Pexip Infinity user experience	37
A beginner's guide to Pexip Infinity	40
Using your Virtual Meeting Room	42

Introduction to Pexip Infinity

Pexip Infinity is a self-hosted, virtualized and distributed multipoint conferencing platform. It can be deployed in an organization's own datacenter, or in a private or public cloud such as Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP) or Oracle Cloud Infrastructure, as well as in any hybrid combination. It enables scaling of video, voice and data collaboration across organizations, enabling everyone to engage in high definition video, web, and audio conferencing.

It provides any number of users with their own personal Virtual Meeting Rooms (VMRs), which they can use to hold conferences, share presentations, and chat. Participants can join over audio or video from any location using the endpoint or client of their choice, including:

- Professional video conferencing room systems (SIP and H.323 devices)
- Desktop/mobile (with the Pexip Infinity Connect suite of clients)
- Web browsers (WebRTC - no downloads required)
- Skype for Business app
- Traditional audio conferencing (PSTN dialing)



Pexip VMRs maintain the same customized address and are always available for spontaneous 1-to-1 or group meetings.

VMRs can also be accessed through a Virtual Reception IVR service, which allows all participants to dial a single number to access Pexip Infinity, and then use the dial tones on their endpoint or phone to select the conference they want to join.

The platform also includes the Infinity Gateway service, which allows end users to place calls to other endpoints that use different protocols and media formats, or to seamlessly connect into an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.

It automatically transcodes all the popular video and audio codecs and supports standard protocols including SIP, H.323, and WebRTC. It supports all standards-based devices including those from Cisco, Poly, Lifesize, Sony, Radvision, Yealink, and Avaya. It

also supports software clients such as Microsoft Skype for Business, Skype for Business Online (Office365), Surface Hub, and Lync 2013.

Scheduling and joining meetings

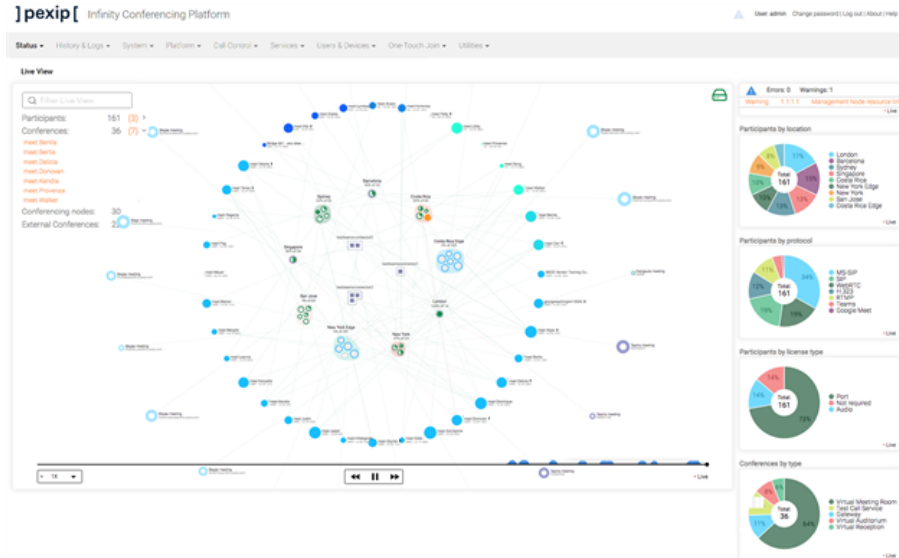
Pexip's [One-Touch Join](#) (OTJ) allows users to schedule a meeting in Microsoft Outlook or Google Calendar and include in the invitation a meeting room with a supported Cisco or Poly videoconferencing endpoint, so that the endpoint in the chosen meeting room displays a Join button just before the meeting is scheduled to begin. Participants can then simply walk into the room and select the button, and the endpoint will automatically dial in to the meeting.

In addition, [VMR Scheduling for Exchange](#) integrates Pexip Infinity with Microsoft Exchange. It enables Microsoft Outlook desktop and Web App users to schedule meetings using Pexip VMRs as a meeting resource.

Management and control

The administrator interface provides a single place to manage the entire Pexip Infinity platform, including the ability to:

- Configure your Pexip Infinity conferencing services (Virtual Meeting Rooms, Virtual Reception and so on).
- Create and manage the Conferencing Nodes that host your conferences.
- Scale on-demand, including dynamic bursting to a cloud service when required, creating capacity only when you need it to save daily running costs.
- Monitor live and historical video usage across the entire platform, as well as call quality and other analytics.
- Perform active conference management tasks such as adding or disconnecting participants, locking a conference, or muting a participant's audio.
- PIN-protect conferences and differentiate between Hosts and Guests to give different users different controls.
- Stream content directly from a VMR to a range of third-party RTMP streaming platforms.



It comes with comprehensive RESTful APIs allowing deep and advanced integrations with numerous services and tools such as PowerBI, plus authentication, authorization and provisioning capabilities against an AD/LDAP database.

You can extend Pexip Infinity's built-in functionality by using external and/or local policy to apply bespoke call policy and routing decisions based on your own specific requirements.

Pexip's unique [distributed architecture](#) is purely software-based and virtualized, running on industry-standard servers, meaning it can be deployed quickly and simply with the flexibility to scale as required. Administrators have access to upgrades as soon as they are released.

Privacy and security

The Pexip Infinity self-hosted solution supports the industry standards for communication encryption for end-user devices, ensuring that all video calls and shared media content is secure and kept private even if it crosses the internet. The entire deployment and all its data, including call status, diagnostic logs and call history, is completely under the ownership and control of the enterprise, even when deployed in the cloud.

In addition, the Pexip Infinity platform has been designed to comply with US Federal security requirements.

Customizable to your standards

You can make Pexip Infinity your own with a variety of [customization options](#). You can easily apply your own branding to the Pexip Infinity platform to produce a personalized user experience.

For more information, see:

- [Components of the Pexip Infinity platform](#)
- [Choosing a deployment environment](#)
- [Pexip Infinity features and specifications](#)

Pexip Infinity features and specifications

The Pexip Infinity platform is designed to use industry-standard servers from any vendor to provide high-quality, scalable and efficient conferencing. The following tables cover the [platform](#), [Infinity Connect](#), [audio and video \(including codecs\)](#), [host hardware](#), [capacity](#) and [hypervisor](#) specifications and requirements.

Pexip Infinity platform


Feature	Description
Application deployment and management	<ul style="list-style-type: none"> • Software-based, virtualized application architecture, running on industry-standard servers. • Management using industry-standard tools, including VMware vSphere, Microsoft Hyper-V, KVM and Xen, and the ability to deploy onto generic hypervisors and orchestration layers. • Ability to deploy on Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP) and Oracle Cloud Infrastructure cloud platforms, including dynamic bursting into Azure, AWS or GCP services when primary conferencing capabilities are reaching their capacity limits. • Integration with the Pexip Private Cloud, where some or all of your transcoding capacity can be hosted privately and securely by Pexip on your behalf. • Flexible deployment model allowing customers to deploy the platform in the way that is most appropriate for them without needing to consume additional software licenses or purchase dedicated hardware. • Ability to seamlessly increase capacity by deploying new, updated, or additional hardware resources. • Management API supporting configuration, status reporting and call control. • Support for Russian and Chinese language in the Pexip Infinity Administrator interface.
Distributed architecture	<ul style="list-style-type: none"> • Efficient distribution to reduce bandwidth consumption over expensive WANs. • Able to deploy dedicated Proxying Edge Nodes to handle all external connections, and leave the conference media processing to privately-addressed Transcoding Conferencing Nodes. • Keeps media as local to each endpoint as possible, reducing the negative impacts of latency, jitter, and packet loss commonly experienced on centralized deployments. • Able to overflow capacity between nodes and locations, providing support for conferences that span multiple physical boxes. • Industry-leading resilience and redundancy capabilities. • A flexible licensing model that allows you to pool conference resources and quickly increase capacity in response to current local requirements.
Intelligent conference management	<ul style="list-style-type: none"> • Upscaling all connected participants to provide a seamless experience to all. • Ability to respond dynamically to fluctuating network conditions by downspeeding and upspeeding individual participants, and support for endpoint-based packet loss recovery and adaptation methodologies (such as packet loss concealment and dynamically adapting bandwidth), thereby protecting the user experience in the event of information loss. • Bandwidth-optimized content sharing towards Infinity Connect clients for crisp image at low bandwidth. • Full support for individual transcoding and transrating of both main stream video and audio, and dual stream content. • Simple conference management and interaction for conference participants using Infinity Connect clients, including the ability for Host participants to add, disconnect, mute and unmute other participants. • Advanced conference management and interaction for administrators (using the web-based Administrator interface or the management API). • Optional tagging of services to allow service providers to track VMR use in CDRs and logs.

Feature	Description
Conferencing services	<ul style="list-style-type: none"> • Virtual Meeting Rooms providing personal meeting spaces for everyone within the organization. • Virtual Auditoriums designed to hold larger lecture-style conferences. • Virtual Reception IVR (Interactive Voice Response) service. • Infinity Gateway enables endpoints to make point-to-point calls to other endpoints that use different protocols and media formats (e.g. from Skype for Business / Lync or WebRTC to H.323). Includes DTMF support. It can also route calls into an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet. • VMR Scheduling for Exchange enables Microsoft Outlook desktop and Web App users to schedule meetings using Pexip VMRs as a meeting resource. • One-Touch Join enables the "click to join" functionality available in VTC endpoints. • VMRs, devices and users can be bulk-provisioned from directory information contained in a Windows Active Directory LDAP server, or any other LDAP-accessible database. • Pexip VMR self-service portal that allows end-users to manage their personal Virtual Meeting Room without having to send requests to their administrator to change the configuration of their VMR. • Choice of layouts: main speaker only; 4 main speakers; main speaker + 7 video thumbnails; main speaker + 21 video thumbnails; 2 main speakers + 21 video thumbnails; or Pexip's AI-driven Adaptive Composition layout featuring real-time automatic face detection and framing. • Conference participants can chat and share messaging content. • Can output a dedicated multimedia stream to enterprise CDN (Content Delivery Network) streaming and recording services such as Wowza, Quickchannel, Qumu, VideoTool, Microsoft Stream and Azure Media Services, and to public streaming services such as YouTube, Facebook and Periscope. • Can integrate with Epic telehealthcare providers. • Ability to manage conferences and participants: <ul style="list-style-type: none"> ◦ PIN-protect conferences and differentiate between Hosts and Guests. ◦ Lock a conference to prevent any further participants from joining. ◦ Transfer a participant to another conference. ◦ Limit the number of participants in a conference, on a per-conference basis. ◦ Limit the bandwidth used by each participant, on a per-conference and/or global basis. • Ability to re-brand with your own images and voice prompts, on a per-conference basis. • Ability to re-brand the Infinity Connect experience. • Ability to integrate Infinity Connect (WebRTC/RTMP) functionality with third-party applications via our client REST API and with websites via the PexRTC library. • Call policy decisions can be taken by an external system or a local policy script. • Test call service that allows users to check their connectivity and the quality of their video and audio.

Feature	Description
Broad interoperability and protocol support	<ul style="list-style-type: none">• Full support for existing industry-standard protocols (SIP, H.323), as well as other technologies (HTML5, Microsoft Lync, Skype for Business, RTMP, WebRTC).• Ability to enable and disable support for individual audio and video codecs.• Easy integration with existing SIP and H.323 call control solutions including Cisco UCM, Cisco VCS, Polycom CMA, Polycom DMA, Avaya Aura, Microsoft Lync 2013, Skype for Business and others.• Integration with Microsoft Teams.• Integration with Google Meet.• Integration with Microsoft Exchange and Office 365.• Conferencing Nodes can act as SIP registrars and as H.323 gatekeepers; nodes in the same system location act as alternate gatekeepers for the purposes of H.323 registration.• Support for automatic call escalation using Multiway (Cisco VCS), call transfer capability (Cisco UCM), and CCCP to a Microsoft Skype for Business / Lync meeting.• Support for presence and customizable avatar published to a Microsoft Skype for Business / Lync client.• Support for automatic dial-out to audio bridges, including automatically issuing conference aliases and pass codes via DTMF tone generation.• IPv4 and IPv6 support.• Support for Far-End Camera Control (FECC).• Support for Cisco One Button to Push (OBTP) and Poly One Touch Dial (OTD).• Ability to tag management, call signaling, and media packets independently with DSCP QoS support.• Support for Forward Error Correction (FEC), downspeeding, bandwidth throttling, and other packet loss concealment technologies.• Unicode support (SIP, Infinity Connect, Administrator interface).
Firewall traversal	<ul style="list-style-type: none">• Static NAT support.• Support for static routes.• Conferencing Nodes can be deployed with dual network interfaces.• Web proxy support.• Far-end NAT traversal (media latching).• Support for media over a TCP connection to assist with firewall traversal.
Security and monitoring	<ul style="list-style-type: none">• Designed to comply with US Federal security requirements.• TLS certificate management, HSTS, certificate signing requests (CSRs).• DTLS support.• Active Directory / LDAP integration for administrator account authentication and authorization.• SNMPv2c and SNMPv3 support.• Support for multiple roles of access.• Authenticated SIP trunks.• Limit Infinity Gateway calls to registered devices only.

Pexip Infinity Connect

Pexip Infinity Connect is a suite of free client software allowing users to connect to Pexip Infinity services from a web browser, installable desktop client, or mobile device.

Feature	Description
Standard features for all Infinity Connect clients	<ul style="list-style-type: none"> • Can be used to join conferences as a full audio/video participant, an audio-only participant, or as a presentation and control-only participant. • Can be used to make point-to-point calls in conjunction with the Infinity Gateway. • Provides conference control to Host participants. • Allows participants to share and view content, whether or not they are connected with video and/or audio. Supported formats are JPEG, BMP, PNG, GIF and PDF. • Infinity Connect desktop client and Infinity Connect web app via Chrome, Opera or Firefox users can share their screen in addition to sharing images and PDFs. • Chat (Instant Messaging) support. • Supports sending of DTMF tones.
Infinity Connect web app	<ul style="list-style-type: none"> • Allows participants to join a Virtual Meeting Room or Virtual Auditorium, or make a call via the Infinity Gateway, using a web browser as their video endpoint. <p>The web app is supported in:</p> <ul style="list-style-type: none"> • Google Chrome version 61 and later (64-bit only) • Mozilla Firefox version 68 and later • Microsoft Edge — all chromium-based versions • Opera version 53 and later • Apple Safari version 11.1 and later on macOS • Apple Safari on iOS 11.2 and later (you must use Safari on iOS devices). <p> We strongly recommend using the latest publicly-released version (i.e. "stable version" or "supported release") of a browser.</p>
Infinity Connect desktop client	<ul style="list-style-type: none"> • Allows a participant to join a Virtual Meeting Room or Virtual Auditorium, or make a call via the Infinity Gateway, using a lightweight client on any PC with any operating system. • Allows users to register their clients in order to receive incoming calls and use directory services. • Can be integrated with Active Directory Federation Services (AD FS), allowing users to register their clients using their AD credentials. <p>Supported on:</p> <ul style="list-style-type: none"> • Microsoft Windows 10 • macOS 10.11 and later • Ubuntu Linux 16.04 and later <p>Note that 32-bit operating systems are not supported with the Infinity Connect desktop client.</p>
Infinity Connect mobile client	<ul style="list-style-type: none"> • Allows a participant to join a Virtual Meeting Room or Virtual Auditorium, or make a call via the Infinity Gateway, using a client downloaded onto their mobile device. • Enables participants to view presentations on their mobile device, regardless of whether they are a video, audio-only, or presentation and control-only participant. <p>Available versions:</p> <ul style="list-style-type: none"> • Infinity Connect mobile client for iOS • Infinity Connect mobile client for Android

Audio and video specifications and codecs

Feature	Description
Supported protocols	<ul style="list-style-type: none"> • H.323 • SIP • WebRTC • RTMP • Microsoft Skype for Business / Lync • Individual protocols can be administratively enabled and disabled.
Audio codecs	<ul style="list-style-type: none"> • G.711(a/μ) • G.719 (this product is covered by patent rights licensed from Telefonaktiebolaget LM Ericsson) • G.722 • G.722.1, G.722.1 Annex C (SIP only) (licensed from Polycom®) • Siren7™, Siren14™ (licensed from Polycom®) • G.729, G.729A, G.729B • Opus • MPEG-4 AAC-LD (MPEG-4 video technology licensed by Fraunhofer IIS) • Speex • AAC-LC
Video codecs	<ul style="list-style-type: none"> • H.261 • H.263, H.263+ • H.264 (Constrained Baseline Profile, Baseline Profile and High Profile), H.264 SVC (UCIF Profiles 0, 1) • VP8 • VP9 (for connections to Conferencing Nodes with processors using AVX2 or later) • RTVideo (licensed from Microsoft®) (deprecated).
Content sharing	<ul style="list-style-type: none"> • H.239 (for H.323) • BFCP (UDP for SIP) • VbSS (for Microsoft Teams and Skype for Business) • RDP (for Microsoft Skype for Business / Lync) • PSOM (for presenting PowerPoint files from Microsoft Skype for Business / Lync clients) • VP8, VP9 (for WebRTC high frame rate) • JPEG (for apps and web).
Bandwidth	<ul style="list-style-type: none"> • Connections from 8 kbps per participant (G.729, audio-only), up to 6 Mbps per participant (will vary depending on the deployment environment, video resolutions, etc).
Other audio and video features	<ul style="list-style-type: none"> • Video resolutions from QCIF to Full HD 1080p (1920 x 1080); 4:3 and 16:9 aspect ratios. • Content resolutions up to 1920 x 1200 (depending on remote side capabilities) • Frame rates up to 30 fps. • Customizable video watermarking. • Pexip StudioSound™ for recording-studio audio quality. • Wideband audio mixing. • Automatic gain control. • Control individual audio via Infinity Connect clients. • Support for AES (128-bit and 256-bit key size), DTLS SRTP, and H.235 for H.323 media encryption.

Host hardware requirements

Feature	Description
CPU	<p>Conferencing Nodes</p> <ul style="list-style-type: none"> We recommend 2nd- or 3rd-generation Intel Xeon Scalable Processors (Cascade Lake / Cooper Lake) Gold 62xx/63xx or 52xx/53xx. We also support Intel Xeon Scalable Processors (Skylake) Gold 61xx generation or E5-2600 v3/v4 Haswell/Broadwell architecture from 2014 or later. Also works with Xeon E5-2600 v1/v2 processors (Sandy Bridge / Ivy Bridge from 2012 or later). AMD processors that support the AVX and AVX2 instruction set are also supported. 2.3 GHz (or faster) clock speed. We recommend 10-20 physical cores per socket. <p>Management Node</p> <ul style="list-style-type: none"> Any processor, 2.0 GHz or faster. 4 cores minimum.
RAM	<p>Conferencing Nodes</p> <p>1 GB RAM per vCPU, so either:</p> <ul style="list-style-type: none"> 1 GB RAM per physical core (if deploying 1 vCPU per core), or 2 GB RAM per physical core (if using hyperthreading and NUMA affinity to deploy 2 vCPUs per core). <p>Management Node</p> <ul style="list-style-type: none"> 4 GB RAM minimum.
Storage	<p>Conferencing Nodes</p> <ul style="list-style-type: none"> 500 GB total per server (to allow for snapshots etc.), including: 50 GB minimum per Conferencing Node <p>Management Node</p> <ul style="list-style-type: none"> 100 GB SSD
GPU	<ul style="list-style-type: none"> Host servers do not require any specific hardware cards or GPUs.
OS	<ul style="list-style-type: none"> The Pexip Infinity VMs are delivered as VM images (.ova etc.) to be run directly on the hypervisor. No OS should be installed.
Network	<ul style="list-style-type: none"> Gigabit Ethernet connectivity is strongly recommended. In general, you can expect 0.5-3 Mbps per call, depending on call control setup.
Multiple VMs sharing the same hardware	<ul style="list-style-type: none"> Pexip Infinity Conferencing Nodes and Management Nodes may share the same physical host. Pexip nodes may also share the same physical host with other virtual machines. Pexip virtual machines must be configured with dedicated CPU and memory resources, i.e. Pexip virtual machines do not support oversubscription.
Service provider considerations	<p>A Pexip deployment can manage multiple customers in various ways:</p> <ul style="list-style-type: none"> Single Management Node, multiple domains, shared Conferencing Nodes A single installation with one Management Node and one or more Conferencing Nodes is used by all customers. Call control or DNS sends calls for all domains to the shared Conferencing Nodes. Does not provide dedicated capacity per customer. Single Management Node, multiple domains, dedicated Conferencing Nodes One or more Conferencing Nodes per customer. Allows for dedicated capacity per customer. Dedicated Management Node and dedicated Conferencing Nodes per customer instance Allows for close customer network integration, using VLANs, hosted on a shared server farm with multiple VLANs. The dedicated Management Node allows for customer self-management.

Capacity

Feature	Description
Call capacity	<p>Capacity is dependent on server specifications. As a general indication, using our recommended hardware (Intel Xeon Gold 6248, 20 cores, 2.5GHz) Pexip Infinity can connect:</p> <ul style="list-style-type: none">• up to two High Definition 720p30 calls per CPU core (based on 1.1 GHz per call plus 20% headroom)• up to 20 audio-only AAC-LD calls at 64 kbps. <p>Servers that are older, have slower processors, or have fewer CPUs, will have a lower overall capacity. Newer servers with faster processors will have a greater capacity. Use of NUMA affinity and hyperthreading will also significantly increase capacity.</p>

Hypervisor requirements

Feature	Description
VMware	<ul style="list-style-type: none">• Version 26 of the Pexip Infinity platform supports VMware vSphere ESXi 6.x.• We recommend at least the Standard edition.• The Enterprise and Enterprise Plus editions have additional features that can be taken advantage of by Pexip Infinity in larger deployments.• The Pexip Infinity platform will run on the free edition of vSphere Hypervisor. However, this edition has a number of limitations that mean we do not recommend its use except in smaller deployments, or test or demo environments.
Microsoft Hyper-V	<ul style="list-style-type: none">• The Pexip Infinity platform supports Microsoft Hyper-V in the form of:• Microsoft Hyper-V Server 2012 and later (including Hyper-V Server 2016)• Windows Server 2012 and later (including Windows Server 2016)
KVM	<ul style="list-style-type: none">• Pexip Infinity requires your KVM environment to include Linux kernel 3.10.0 or later, and QEMU 1.5.0 or later. This means the following distributions: Debian 8, RHEL 7, SLES 12, or Ubuntu 14.04 (or later, where appropriate).
Xen	<ul style="list-style-type: none">• Pexip Infinity requires Xen 4.2 and later.
Other hypervisors and orchestration layers	<ul style="list-style-type: none">• Conferencing Nodes can be provisioned with a configuration document generated independently of a generic VM image. This permits deployment of Pexip Infinity onto unsupported hypervisors as well as onto supported hypervisors that are managed by an orchestration layer.• Pexip Infinity can be deployed on Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP) or Oracle Cloud Infrastructure, and on the HPE Helion Openstack® Cloud platform.



What's new in version 26?

The [new features and enhancements](#) and [changes in functionality](#) included in Pexip Infinity version 26, along with any [planned changes](#) and the [new features and changes in the Infinity Connect web app](#) are described below.


For full information about this release, see the [release notes](#).

For information about earlier versions of Pexip Infinity, see [Features added in previous releases](#).

Pexip Infinity new features and enhancements

Feature	Description	More information
Adaptive Composition layout enhancements for inactive video participants and Virtual Auditoriums	<p>The Adaptive Composition layout contains improvements to the process of removing and restoring inactive video participants in the video mix:</p> <ul style="list-style-type: none"> There is a new inactive video participants indicator . This shows the number of inactive video participants who are excluded from the video layout, including any Infinity Connect participants who have muted their video. Previously these participants were included in the count of audio-only participants. Inactive participants are more quickly returned to the video layout when they are detected or become active again, removing the need for having to speak to be returned to the mix. <ul style="list-style-type: none"> Web app participants: any activity in the web app, such as speaking or moving the mouse, instantly triggers a face-detection check. Endpoint participants: now performs a face-detection check every 30 seconds, as well as when the participant speaks. <p>Other enhancements in this release:</p> <ul style="list-style-type: none"> The Adaptive Composition layout can now be used in Virtual Auditoriums (in addition to Virtual Meeting Rooms). Support for Adaptive Composition in Pexip's Cloud Video Interop (CVI) integration with Microsoft Teams (see below). 	Selecting the layout seen by participants
Removal of video-muted Infinity Connect participants from other layouts	<p>Other (non-Adaptive Composition) layouts now also remove Infinity Connect participants from the layout mix if they mute their video:</p> <ul style="list-style-type: none"> These are shown with a separate  indicator on the left-hand side of the screen, and works in the same manner as for existing audio-only participant indicators. Up to 4 audio-only or muted-video indicators are displayed before they are collapsed into a single indicator. 	Selecting the layout seen by participants

Feature	Description	More information
Teams Connector enhancements: Teams lobby notifications, Azure resource tagging, and support for Adaptive Composition*	<p>Pexip's Cloud Video Interop (CVI) integration with Microsoft Teams has been enhanced:</p> <ul style="list-style-type: none"> VTC systems that are gatewayed into a Teams meeting now see notification messages when somebody is waiting in the Teams lobby: <ul style="list-style-type: none"> The messages are for information only — the VTC system cannot admit people into the meeting. The notifications use Pexip Infinity's existing locked conference indicators, so the messages and indicators can be customized via themes in the usual way. <p>Other changes and improvements to the Teams Connector include:</p> <ul style="list-style-type: none"> The ability to optionally specify a set of tags (name-value pairs) to apply to the Azure resources that are created for the Teams Connector. This uses a new <code>\$tags</code> variable in the PowerShell variables initialization script. The redeploy/upgrade script now removes the dynamic Azure resource groups for you (those named <code><prefix>-TeamsConn-<region>-RG</code> that contain the existing Teams Connector instances). This means you no longer have to perform this step manually via the Azure portal when upgrading or redeploying. Support for Adaptive Composition layout (technology preview) <p>When Adaptive Composition is used with a Teams conference:</p> <ul style="list-style-type: none"> Up to 9 video participants are shown in the Adaptive Composition layout seen by VTC participants. Each participant's video that is received from Teams for display to VTC participants is cropped and framed as appropriate. The VTC participant's video stream sent to Teams is cropped and framed as appropriate. <p>The default layout for Teams meetings is still Pexip's standard 1+7 layout, but you can change this to Adaptive Composition via policy or via the transform layout management or client API commands.</p> <ul style="list-style-type: none"> Version 26 of the Teams Connector contains updates that necessitate an upgrade to your Pexip platform to ensure compatibility with the latest updates to the Microsoft Teams APIs and to the Teams Connector's latest features. <p>We strongly recommend that you upgrade your Pexip deployment — both the Pexip Infinity platform and the Pexip Teams Connector — to version 26 as soon as practicable.</p>	
WebRTC quality improvements	<p>This release contains the following WebRTC enhancements:</p> <ul style="list-style-type: none"> Improved video and audio resilience in lossy networks. Improved resilience and image quality for full-motion and still image presentation (now uses H.264). 	
Epic telehealth patients can join via email or SMS	<p>Pexip's Epic telehealth integration contains the following new features:</p> <ul style="list-style-type: none"> Epic telehealth patients can now launch their video visit via a join link sent by email or SMS text message. Pexip Infinity administrators can configure how the conference/appointment name is constructed via a new Service name template profile field. Pexip Infinity administrators can configure the content of the error page shown to users if a telehealth call launch fails via a new Error page template for launch failures profile field. 	

Feature	Description	More information
Polling for scheduled maintenance events in Microsoft Azure	Pexip Infinity VMs that are deployed in Microsoft Azure now poll for scheduled maintenance events automatically. When certain events are detected, such as a Freeze event, the Conferencing Node running on that VM is automatically placed into maintenance mode, and an alarm is raised.	
Additional One-Touch Join meeting rules	<ul style="list-style-type: none"> There is a new One-Touch Join rule available to support invitations to Microsoft Teams meetings sent from domains other than your own, where the meeting organizer is using a Cisco — Teams integration. There is a new option for Zoom meetings to include the meeting password in the alias used to join the meeting, so that users do not need to enter the password themselves. 	One-Touch Join meeting types and transforms
New 1 + 33 conference layout *	<p>A new 1 + 33 layout is available.</p> <p>This layout is suitable for large conferences. It displays 1 small main speaker and up to 33 other participants. (It can be enabled for selection via Platform > Global Settings > Tech Preview Features).</p>	Selecting the layout seen by participants
Administrative improvements	<p>This release contains the following administrative improvements:</p> <ul style="list-style-type: none"> There is a new <code>icon_video_muted.svg</code> graphic  (video_muted_indicator.png in legacy version 1 themes), which is used in non-Adaptive Composition layouts when a muted video participant is excluded from the video layout. In license management, if you select an existing license there is now an option to manually return the license (in addition to the existing online return option). Quicker reboot times when restarting a Conferencing Node on the same hardware due to faster sampling. 	Rules and requirements for customized themes Pexip Infinity license installation and usage

* Technology preview only

Pexip Infinity changes in functionality

Feature	Description	More information
HTTP Content-Security-Policy headers on Conferencing Nodes are now enabled by default, and other new security wizard options	<p>The use of HTTP Content-Security-Policy (CSP) headers on Conferencing Nodes has changed:</p> <ul style="list-style-type: none"> CSP headers are now enabled by default on new and upgraded deployments (previously disabled by default). You can now enable or disable HTTP Content-Security-Policy headers, and modify the header contents, via the Administrator interface (Platform > Global Settings > Security). The option in the security wizard to enable/disable CSP headers has been removed. The default CSP header content is unchanged from previous releases. The security wizard also contains the following new options: <ul style="list-style-type: none"> Enable Referrer-Policy: same-origin on HTTP responses (default = yes) Enable X-Frame-Options: DENY (yes) Enable preload in HSTS header (no) 	About global settings



Feature	Description	More information
Break-in resistance settings: allow list and changes to the handling of X-Forwarded-For headers	<p>Pexip Infinity version 26 behaves differently from prior versions in its handling of X-Forwarded-For headers in its break-in resistance checking.</p> <p>Installations that use reverse proxies in conjunction with the break-in resistance options in Pexip Infinity need to take action after upgrading to v26:</p> <ul style="list-style-type: none"> A new option in version 26 lets you define "Allow list address" entries that are exempt from break-in resistance checks (Call Control > Break-In Attempt Allow List). You must create an allow list address with an Entry type of <i>Proxy</i> that contains the apparent IP address (from Pexip Infinity's perspective) for all trusted reverse proxies or HTTP(S) load balancers in use in your deployment. This allows Pexip Infinity to process the X-Forwarded-For header entries and, in turn, means that the break-in resistance features will work correctly to block malicious attackers who attempt to attack the Pexip Infinity deployment via the reverse proxy. You may also choose to add trusted call control systems (such as H.323 gatekeepers and SIP proxies e.g. Cisco VCS systems) to the allow list to prevent those IP addresses from ever being temporarily blocked by Pexip Infinity. <p>If you have previously disabled the break-in resistance features due to the attacks being blocked too aggressively, you may find that you can now find a balanced configuration that allows you to enable the feature and maintain a good level of protection against SIP scanners etc. but, with judicious use of the new Allow List feature, a significantly reduced risk of accidentally blocking legitimate users from accessing Pexip Infinity.</p>	Break-in resistance settings to mitigate rogue calls
New permission when configuring dynamic bursting to the Microsoft Azure cloud	<p>When setting up the Active Directory (AD) application and permissions to be used by Pexip Infinity to start up and shut down the Conferencing Node overflow instances, we now recommend using the <i>Virtual Machine Contributor</i> built-in role (previously <i>DevTest Labs User</i>).</p> <p>If you have created your own custom role you should add the <code>Microsoft.Compute/virtualMachines/powerOff/action</code> permission to it.</p> <p>If the <code>powerOff</code> action permission is not assigned you may temporarily see a "Not authorized to perform this operation" alarm, but the virtual machine will still get successfully deallocated.</p>	
Administrative modifications	<p>This release contains the following administrative modifications:</p> <ul style="list-style-type: none"> You no longer have to select a Telehealth Profile when configuring a Call Routing Rule for Epic telehealth integrations (this field previously had no effect and was just for your own information). The rule's call target option has also been renamed from <i>Epic Telehealth profile</i> to <i>Epic Telehealth meeting</i>. 	Configuring Pexip Infinity to integrate with Epic telehealth
Support for old versions of ESXi	Support for ESXi 5.x was removed in Pexip Infinity v26. Version 26 of the Pexip Infinity platform supports VMware vSphere ESXi 6.x.	
Automatic deployments on ESXi no longer supported	Automatic deployment of Conferencing Nodes in VMware environments was deprecated in Pexip Infinity v23 and was removed in v26. From this version onwards, you must deploy your Conferencing Nodes manually , as with other hypervisor environments. As a consequence, VM managers are no longer required or supported.	

Planned changes in future releases

Feature	Description	More information
RTVideo codec, Lync Server 2010 and Lync 2010 clients no longer supported	Technical support for the RTVideo codec is deprecated since Pexip Infinity v25. The RTVideo codec will be removed completely in a future release which will then disable interoperability with Lync Server 2010 and Lync 2010 clients.	

Infinity Connect web app new features and changes

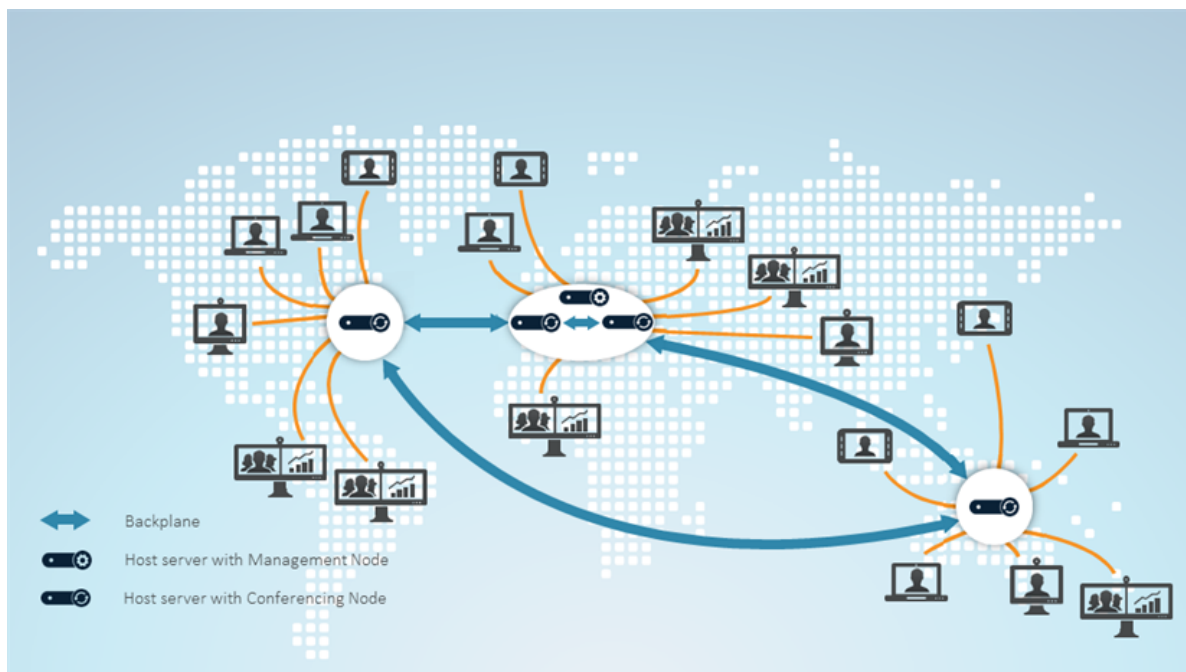
Following are the new features and changes in the Infinity Connect web app in Pexip Infinity version 26:

Feature	Description	More information
New audio-only and inactive-video indicators in the participant list	<p>Each person in the Participant list now has an additional icon displayed if that participant is connected as audio-only, is a video-muted Infinity Connect participant, or is inactive, for example, if they are away from the computer (and thus their face cannot be detected in their video stream):</p> <div> <div>Alice</div> <div>Bob</div> <div>Randolph</div> </div> <p>where the indicators represent:</p> <div>  Muted video or inactive (away)  Connected via audio only </div>	
More bandwidth options in the app settings	<p>There are more bandwidth selection options available in Settings > Bandwidth. You can now choose from <i>Very Low (up to 256kbps)</i>, <i>Low (up to 576kbps)</i>, <i>Medium (up to 1264kbps)</i>, <i>High (up to 2464kbps)</i>, or <i>Very High (up to 6144kbps)</i>.</p> <p>The default is <i>Medium (up to 1264kbps)</i>.</p>	
Enable "raise hand" in VMRs	By default the "raise hand" feature is only available in Virtual Auditoriums. There is a new <code>raiseHandInVMR</code> customization option in the <code>settings.json</code> file in a web app branding package that can be used to also enable the feature in Virtual Meeting Rooms.	

Components of the Pexip Infinity platform

The Pexip Infinity conferencing platform is a virtual entity that consists of a **Management Node** and one or more securely interconnected **Conferencing Nodes**. Both are software applications that you deploy as Virtual Machines (VMs) on **host servers** distributed around the globe, or via a **cloud service**. You can add, remove or move Conferencing Nodes according to your conferencing capacity requirements.

- Conferences take place in **Virtual Meeting Rooms** and **Virtual Auditoriums**, with each having one or more associated **aliases**. Conference participants access a Virtual Meeting Room or Virtual Auditorium by dialing any one of its aliases directly, or via the **Virtual Reception** IVR service. This connects them to the Virtual Meeting Room or Virtual Auditorium on their nearest Conferencing Node. A single such conference can take place across one, two, or more Conferencing Nodes with no difference in conference experience from the participants' perspective.
- Conference participants can access Virtual Meeting Rooms and Virtual Auditoriums from virtually any endpoint, including the **Pexip Infinity Connect** suite of clients (which includes a desktop client, mobile clients and a web app). Infinity Connect clients can also be used to control the conference, view presentations, share content, and chat with other conference participants.
- The **Pexip Infinity Distributed Gateway** allows users to make person-to-person calls between virtually any type of endpoint (including SIP and H.323 devices, Skype for Business / Lync, and Pexip's own Infinity Connect clients). It also enables these endpoints to join an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.
- **VMR Scheduling for Exchange** enables Microsoft Outlook desktop and Web App users (using Office 365, Exchange 2013 or Exchange 2016) to schedule meetings using Pexip VMRs as a meeting resource.
- Pexip **One-Touch Join** integrates support for videoconferencing endpoints' "click to join" workflows. It can be integrated within an existing or new Pexip Infinity deployment, or installed as a stand-alone deployment with its own dedicated Management Node and Conferencing Nodes.
- The Pexip VMR self-service portal is a separately-installable component that allows end-users to manage their personal Virtual Meeting Room without having to send requests to their administrator to change the configuration of their VMR.



Pexip Infinity deployment showing Management Node and four Conferencing Nodes with participants connected locally

Management Node

The Management Node is the administrative interface of the Pexip Infinity platform, from which administrators can:

- Create and manage Conferencing Nodes.
- Configure Pexip Infinity services (Virtual Meeting Rooms, Virtual Receptions and so on).
- View platform and conference status across all Conferencing Nodes.

- Perform active conference management functions such as adding and disconnecting participants, enabling streaming or recording services, locking a conference, or muting a participant's audio.

The Management Node does not handle any conference media or signaling.

It is deployed using a virtual machine management application such as VMware's vCenter Server, or Microsoft Hyper-V, or on a cloud service such as Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP) or Oracle Cloud Infrastructure.

Conferencing Nodes

The Conferencing Nodes provide the capacity for conferences.

- They handle all conference media and signaling.
- A Conferencing Node can have either a transcoding or a proxying role:
 - Transcoding Conferencing Nodes are required in all deployments; they manage all of the media processing required to host a conference, and can also handle direct connections to/from endpoints if required.
 - Proxying Edge Nodes are optional; they handle call signaling and the media connection with the endpoint, but forward the media on to a Transcoding Conferencing Node for processing. For more information, see [Distributed Proxying Edge Nodes](#).
- There is no limit on the number of Conferencing Nodes that you can add to the Pexip Infinity platform.
- All Conferencing Nodes get the same service configuration from the Management Node. This means that participants throughout your organization can access the same Pexip Infinity services (Virtual Meeting Rooms, Virtual Receptions and so on) even though they might be connected to different Conferencing Nodes.
- Conferencing Nodes are deployed via the Management Node. You use the Management Node to configure the new Conferencing Node and generate a configuration file, then complete the deployment using the appropriate hypervisor or cloud-provider tools.
- The Pexip Infinity platform can have Conferencing Nodes that are deployed on one or more host servers, across one or more system locations and managed by one or more types of hypervisor, or it can be a hybrid deployment with nodes running on a combination of on-premises and cloud-hosted servers. A Conferencing Node can co-exist on the same host server as a Management Node.
- Conferencing Nodes can be deployed with dual network interfaces.

Pexip Infinity Connect clients

Conference participants do not need to have a traditional video endpoint in order to access Pexip Infinity services.

The complementary Pexip Infinity Connect suite of clients allows users to connect to any conference, either:

- directly from a web browser without any special downloads or plugins
- from an installable desktop client
- from a mobile client, available for iOS or Android.

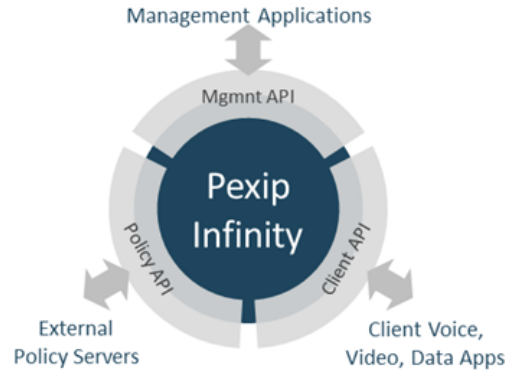
In addition to connecting with video and audio, Infinity Connect users can control the conference, view presentations, share content and chat. Infinity Connect can also be used to make direct calls to other devices or systems when used in conjunction with the Infinity Gateway.

For more information on using and administering Infinity Connect, see [Introduction to Infinity Connect](#).

Pexip Infinity APIs and SDKs

Pexip Infinity incorporates several powerful and comprehensive APIs:

- **Management API:** a REST API used for configuring the entire Pexip Infinity deployment, viewing history and status, and issuing commands. See [Introduction to the management API](#) for more information.
- **Client API:** a REST API used for managing calls and participants, such as connect, disconnect, mute and unmute, presentation controls, DTMF, etc. See [Pexip client REST API](#) for more information.
- **Policy API:** a REST API used to defer decision-making to external policy servers instead of using the built-in call policies within Pexip Infinity. See [Using external and local policy to control Pexip Infinity behavior](#) for more information.



In addition to these REST APIs, a Javascript API is also available for building custom web-based clients. See [PexRTC JavaScript client API](#) for more information.

Conference types and services

The Pexip Infinity platform offers a variety of conference types and services:

- [Virtual Meeting Rooms](#) and [Virtual Auditoriums](#) are used to hold conferences, share presentations, and chat. Participants can join over audio or video from any location using virtually any type of communications tool, such as Microsoft Skype for Business / Lync, a traditional conferencing endpoint, a mobile telephone, or a Pexip Infinity Connect client.
- The [Virtual Reception](#) IVR service provides a way for conference participants who cannot dial Virtual Meeting Room and Virtual Auditorium aliases directly, to access these services from a central point using DTMF tones. It can also be used to route calls via the Infinity Gateway.
- The [Pexip Infinity Distributed Gateway](#):
 - Enables endpoints within your deployment to make direct calls to other endpoints. As with calls into VMRs, the gateway can interwork the protocols and media formats used by each type of device (SIP, H.323, WebRTC etc).
 - Enables any type of endpoint, including traditional VTC endpoints, to join externally-hosted meeting services such as Skype for Business / Lync meetings, Microsoft Teams, and Google Meet,
 - Can be used with call control systems and other third party services to enable calls from your deployment to external devices including PSTN and mobile phones.
- [VMR Scheduling for Exchange](#) integrates Pexip Infinity with Microsoft Exchange. It enables Microsoft Outlook desktop and Web App users (using Office 365, Exchange 2013 or Exchange 2016) to schedule meetings using Pexip VMRs as a meeting resource.
- Pexip's [One-Touch Join](#) (OTJ) allows users to schedule a meeting in Microsoft Outlook or Google Calendar and include in the invitation a meeting room with a supported Cisco or Poly videoconferencing endpoint, so that the endpoint in the chosen meeting room displays a Join button just before the meeting is scheduled to begin. Participants can then simply walk into the room and select the button, and the endpoint will automatically dial in to the meeting.
- A [Test Call Service](#) provides a test loopback service that allows users to check the quality of their video and audio (i.e. that their local camera, microphone and speakers are working properly), and verifies that they can connect to a Conferencing Node.

VMR self-service portal

The Pexip VMR self-service portal is a separately-installable component that allows end-users to manage their personal Virtual Meeting Room without having to send requests to their administrator to change the configuration of their VMR.

Pexip provides the VMR portal appliance via an OVA template suitable for deployment on VMware ESXi. The OVA template is provided "as-is" and provides a reference installation which is suitable for integrating with an existing Pexip Infinity deployment.

Host servers

The Management Node and Conferencing Nodes are virtual machines (VMs) that run on industry-standard host servers. A Management Node can run on the same host server as a Conferencing Node. Other Conferencing Nodes can run on host servers in the same or different locations, allowing you to create a globally distributed system.

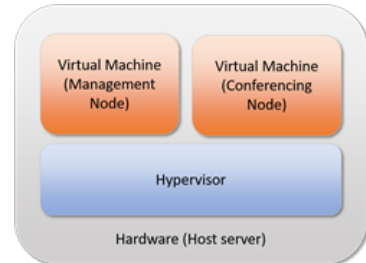
You can have two Conferencing Nodes running on the same host server, for example to ensure service continuity during upgrade of one of the Conferencing Nodes. However, you must ensure that your hardware is not over-committed - see [Detailed server hardware requirements](#) for more information.

The Pexip Infinity platform can also be deployed as a [cloud service](#) via **Amazon Web Services (AWS)**, **Microsoft Azure**, **Google Cloud Platform**, or **Oracle Cloud Infrastructure**, with private, public or hybrid deployment options.

Hypervisors

Each host server runs a hypervisor, an application which manages virtual machines and the physical hardware on which they are hosted. Pexip Infinity version 26 includes specific support for the following hypervisors:

- VMware vSphere ESXi (6.x)
- Microsoft Hyper-V Server 2012 and later (including Hyper-V Server 2016); Windows Server 2012 and later (including Windows Server 2016)
- KVM
- Xen (4.2 and later)



Other hypervisors and orchestration layers may be used but are not officially supported. If you wish to deploy Pexip Infinity using a non-supported hypervisor, we recommend that you contact your Pexip authorized support representative for assistance.

Call control

Supported call control solutions

Pexip Infinity can be easily integrated with virtually any existing SIP, H.323 and Skype for Business / Lync call control solutions including Cisco UCM, Cisco VCS, Polycom CMA, Polycom DMA, Avaya Aura, Microsoft Lync 2013, Skype for Business, and others.

Local and external policy

You can extend Pexip Infinity's built-in functionality by using external and/or local policy to apply bespoke call policy and routing decisions based on your own specific requirements.

See [Using external and local policy to control Pexip Infinity behavior](#) for more information.

Endpoint registrations

Pexip Infinity can act as a SIP registrar and H.323 gatekeeper, which means that you can register SIP and H.323 endpoints directly to Pexip Infinity. This allows Pexip Infinity to route calls to those registered devices without having to go via an external SIP proxy or H.323 gatekeeper, or rely on DNS.

Infinity Connect desktop clients and legacy versions of the Infinity Connect mobile clients for Android can also register to Pexip Infinity Conferencing Nodes. This allows these devices to receive calls via Pexip Infinity and use directory lookup services.

For more information, see [Registering devices to Pexip Infinity](#) and [DNS record examples](#).

Note that the Pexip Infinity platform does not register with external gatekeepers as an MCU.

Using an external gatekeeper to route calls to Pexip Infinity conferences

To ensure that calls can be routed to Pexip Infinity, your gatekeeper or call control system must be configured with appropriate neighbor/zone relationships towards the Pexip Infinity Conferencing Nodes. These zones must be set up so that when an endpoint places a call to a Pexip Infinity alias, the call is routed to the endpoint's local Conferencing Node(s) as a first preference. Other non-local Conferencing Nodes can be used as secondary choices to provide redundancy.

For further information about how to configure your specific call management system to work with Pexip Infinity, see the following documentation:

- [Pexip Infinity and Microsoft Skype for Business / Lync Deployment Guide](#)
- [Pexip Infinity and Cisco VCS Deployment Guide](#)

- [Pexip Infinity and Cisco Unified Communications Manager Deployment Guide](#)
- [Pexip Infinity and Polycom DMA Deployment Guide](#)

Distributed architecture

Benefits of the Pexip Infinity distributed architecture

Pexip Infinity is built on top of a distributed architecture which provides:

- Centralized management of any number of Conferencing Nodes in any number of locations.
- Ability to deploy conferencing resources where and when required, without service outage.
- Ability to deploy dedicated Proxying Edge Nodes to handle all external connections, and leave the conference media processing to privately-addressed Transcoding Conferencing Nodes.
- Significant WAN bandwidth savings in conferences that span locations.
- Consistent user experience, independent of the number of Conferencing Nodes.
- Increased resilience to temporary network outages.
- Ability to use Conferencing Nodes as gateways for person-to-person calls, thus avoiding hairpinning of media back to a centralized datacenter.
- Allocation of licenses from a central pool.

Distributed architecture components

Pexip Infinity distributed architecture is purely software-based and virtualized, running on industry-standard servers. It consists of:

- A single Management Node. A Pexip Infinity deployment, regardless of size, has just one Management Node. The purpose of the Management Node is to create and manage Conferencing Nodes. The Management Node is in neither the signaling nor the media path of a conference.
- One or more Conferencing Nodes. Conferencing Nodes handle all aspects of the media and signaling connections to endpoints and other devices, and host the associated conferences. A minimal Pexip Infinity deployment has one Management Node and one Conferencing Node. However, most Pexip Infinity deployments will have multiple Conferencing Nodes.

A Management Node can run on the same host server as a Conferencing Node. Other Conferencing Nodes can run on host servers in the same or different locations, allowing you to create a globally distributed system. You can have two Conferencing Nodes running on the same host server, for example to ensure service continuity during upgrade of one of the Conferencing Nodes, and for maximum performance.

The Pexip Infinity platform can also be deployed as a [cloud service](#) via **Amazon Web Services (AWS)**, **Microsoft Azure**, **Google Cloud Platform**, or **Oracle Cloud Infrastructure**, with private, public or hybrid deployment options.

Centralized management

Configuration and provisioning data is pushed out from the Management Node to all the Conferencing Nodes, and diagnostics data and event information is sent from the Conferencing Nodes back to the Management Node. Administrators never have to manage a Conferencing Node directly. This centralized management ensures that the entire deployment is configured with a consistent data set – the entire deployment acts as a single application.

Scaling up

You can easily scale a deployment up by creating several Conferencing Nodes in the same location (i.e. the same datacenter). Capacity can even be added “on the fly” – Conferencing Nodes can be added in a couple of minutes if more capacity is needed. Alternatively, each location can be configured to overflow to another location if it reaches its capacity, including bursting to temporary resources on a cloud service.

Scaling out

A typical Pexip Infinity deployment consists of two or more locations. If a customer has three main offices such as New York, London and Tokyo, and a concentration of users in those locations, you would typically deploy Conferencing Nodes in all three locations. There is no limit on the number of locations in a Pexip Infinity deployment. Additional locations can be added “on the fly” while the system is operating, with no impact on service availability.

Application level resiliency

Application level resiliency greatly improves on a conference experience during for instance temporary network outages, as Pexip Infinity will automatically re-establish the conference when the network connection is re-established.

Conference distribution

All Pexip Infinity services (Virtual Meeting Rooms, Virtual Auditoriums, Virtual Receptions and the Infinity Gateway) can be accessed via any Conferencing Node. When a user dials in to a Virtual Meeting Room, or Virtual Auditorium a **conference instance** is created. As more users dial in to the conference, it may be managed across one or more Conferencing Nodes.

When you deploy a Conferencing Node, you select its system location and role:

- **System location:** this is normally used within Pexip Infinity to group together those nodes that are in the same physical location. Conferences that span more than one Transcoding Conferencing Node can be [locally distributed](#), [globally distributed](#), or [both](#), depending on the system location of each of the nodes involved. These various distribution scenarios are described in more detail below.
- **Role:** determines if the Conferencing Node has a **transcoding** role (i.e. it is a Transcoding Conferencing Node that manages and hosts conferences) or has a **proxying** role (i.e. it is a Proxying Edge Node that forwards the client's media onto a Transcoding Conferencing Node). Proxying Edge Nodes (and their associated locations) do not affect whether a conference is locally or globally distributed, as they simply forward media onto the Transcoding Conferencing Nodes that are responsible for hosting the conference — but the locations of the transcoding nodes that process the media do affect how the conference is distributed. See [Conference distribution and Proxying Edge Nodes](#) for an example of calls received on proxying nodes that are forwarded onto transcoding nodes in a locally and geographically distributed conference.

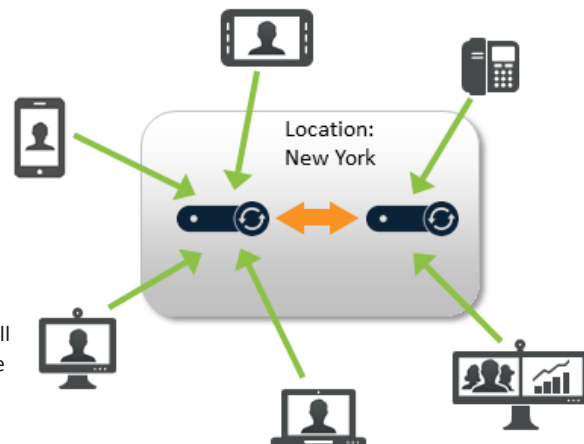
When two or more Transcoding Conferencing Nodes are hosting the same conference, they send the call media between each other over a secure IPsec **backplane**.

Locally distributed conferences

Locally distributed conferences exist across multiple Transcoding Conferencing Nodes in the same **System location**, as shown in the diagram opposite.

Having more than one transcoding node in a single location allows you to increase conferencing capacity and provide redundancy.

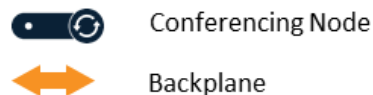
- i** To maintain efficiency, no more than three transcoding nodes **per location** will handle the media for a **particular conference instance**. However, you can configure "overflow" locations that will handle the media if a location reaches its capacity for a conference instance. For more information, see [Handling of media and signaling](#).



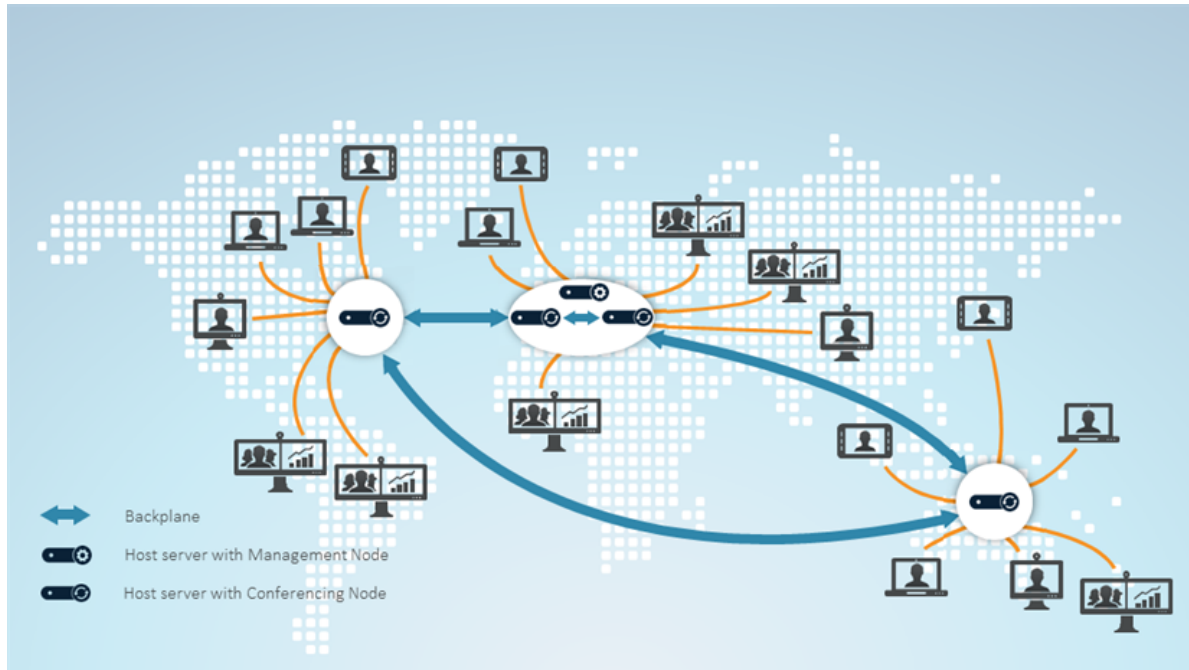
Globally distributed conferences

Globally distributed conferences exist across several Transcoding Conferencing Nodes, where each node is in a different system location. As system locations are typically used to represent different physical locations, this allows participants in different regions to access the conference from their local Conferencing Node. The nodes send the call media for the conference to each other over a single geo backplane, with each node sending the media on behalf of all the endpoints connected (or proxied) to it, thus minimizing WAN bandwidth usage between locations.

This helps to provide a superior meeting experience as it reduces the distance that the media has to travel between the endpoint / client and the Conferencing Nodes.



Locally and globally distributed conferences



Locally and globally distributed conference with four Conferencing Nodes across four host servers in three locations

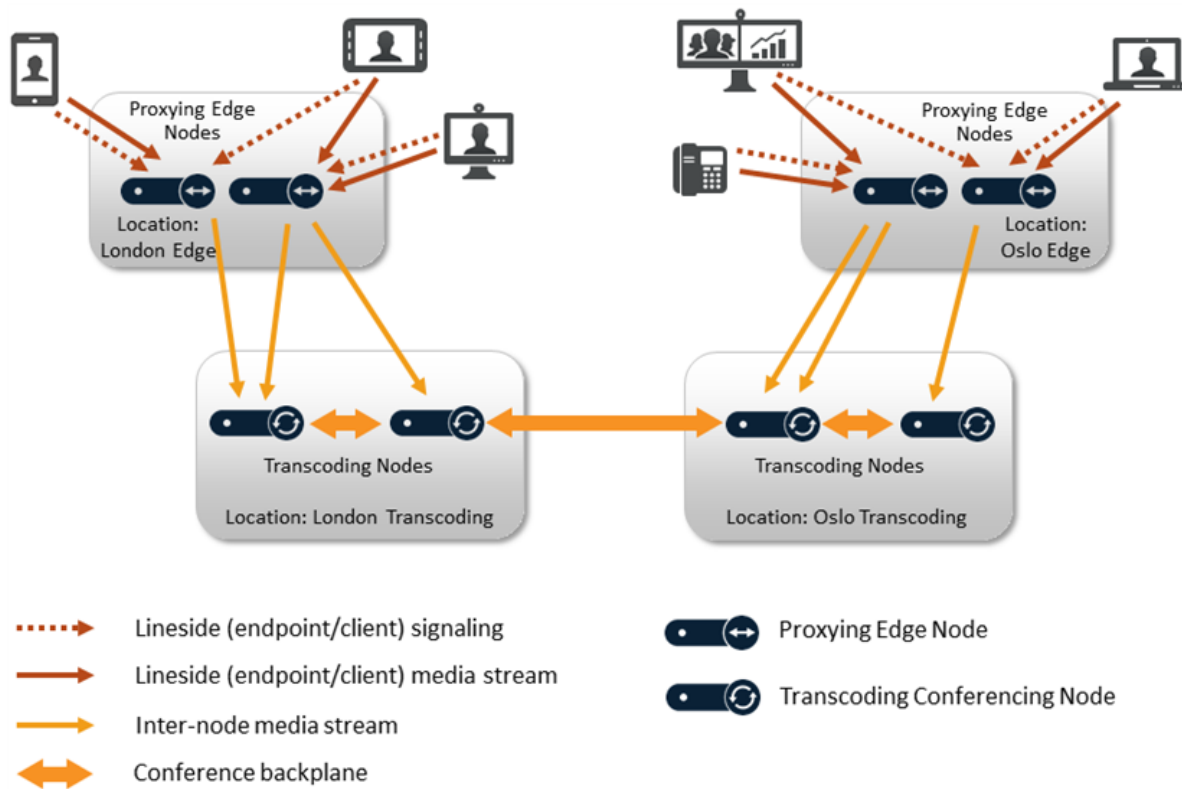
A conference can be **locally and globally distributed** at the same time, if two or more Transcoding Conferencing Nodes in one location and at least one other transcoding node in a different location are involved. In such cases, one transcoding node in each location acts as the intermediary for any other transcoding nodes in the same location that are handling the media for that conference. Call media for each location is sent between the intermediaries only, thus minimizing WAN bandwidth usage between locations.

Conference escalation from locally to globally distributed is handled automatically by Pexip Infinity and is seamless to conference participants.

Conference distribution and Proxying Edge Nodes

Proxying Edge Nodes (and their associated locations) do not affect whether a conference is locally or globally distributed, as they simply forward media onto the Transcoding Conferencing Nodes that are responsible for hosting the conference — but the locations of the transcoding nodes that process the media do affect how the conference is distributed.

A system location should not contain a mixture of proxying nodes and transcoding nodes. Hence, in the example scenario shown here, the Conferencing Nodes in the two locations "London Edge" and "Oslo Edge" are Proxying Edge Nodes and thus those nodes and locations are not involved in the actual hosting of any conferences. They forward the media onto the Transcoding Conferencing Nodes in the "London Transcoding" and "Oslo Transcoding" locations respectively. This conference is locally distributed within both of those "transcoding" locations (as there are multiple nodes hosting the conference within each location, with a local backplane between those nodes), and as the conference is split across two locations, it is also geographically distributed and therefore there is a geo backplane between one of the nodes in the "London Transcoding" location and one of the nodes in the "Oslo Transcoding" location.



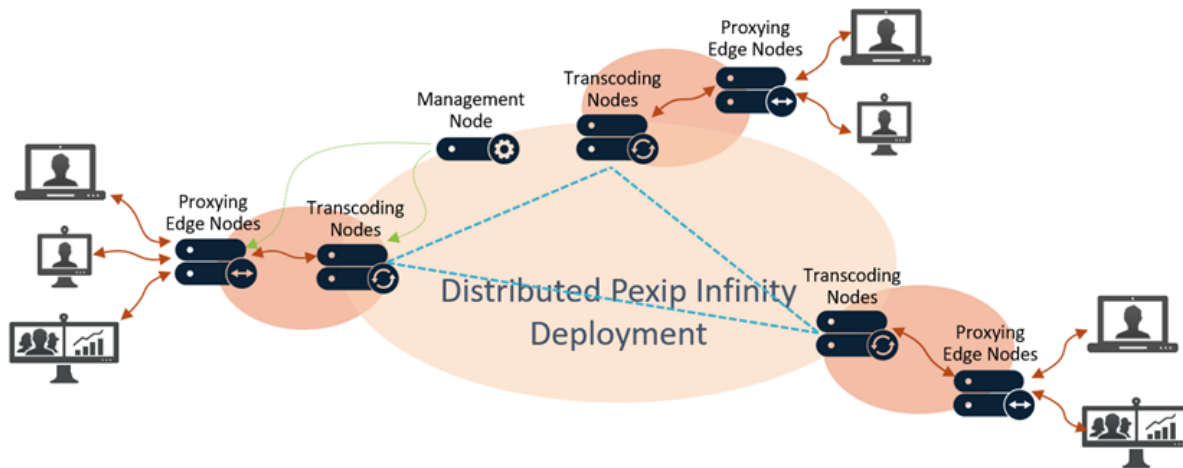
Distributed Proxying Edge Nodes

When designing your Pexip Infinity deployment, one of the main considerations is how and where to deploy your Conferencing Nodes. A Conferencing Node can perform one of two roles:

- **Proxying:** a Proxying Edge Node handles all media and signaling connections with an endpoint or external device, but does not host any conferences — instead it forwards the media on to a Transcoding Conferencing Node for processing.
- **Transcoding:** a Transcoding Conferencing Node handles all the media processing, protocol interworking, mixing and so on that is required in hosting Pexip Infinity calls and conferences. When combined with Proxying Edge Nodes, a transcoding node typically only processes the media forwarded on to it by those proxying nodes and has no direct connection with endpoints or external devices. However, a transcoding node can still receive and process the signaling and media directly from an endpoint or external device if required.

You can deploy your Pexip Infinity platform as either a mix of Proxying Edge Nodes and Transcoding Conferencing Nodes, or as a system that only contains Transcoding Conferencing Nodes.

A typical deployment scenario is to use Proxying Edge Nodes as a front for many privately-addressed Transcoding Conferencing Nodes. Those outward-facing proxying nodes would receive all the signaling and media from endpoints and other external systems, and then forward that media onto other internally-located transcoding nodes to perform the standard Pexip Infinity transcoding, gatewaying and conferencing hosting functions.



How it works

When you deploy a new Conferencing Node, you must decide on its role — either **transcoding** or **proxying** (although you can change its role later).

- If an endpoint connects to a Transcoding Conferencing Node, that node will handle the signaling connection with that endpoint, and (subject to media allocation rules) it may also host the associated conference or gateway call on that node.
- If an endpoint connects to a Proxying Edge Node, that node will still handle the signaling connection to the endpoint, but it (or another proxying node in that location) will then proxy the media on to a Transcoding Conferencing Node for processing. The proxying node will perform any encryption or decryption of the media stream that may be required.

The benefits of using Proxying Edge Nodes to handle all connections from endpoints and other systems — and leaving the media processing to internally-located Transcoding Conferencing Nodes — are:

- You only need to deploy certificates on your Proxying Edge Nodes — only those nodes that handle the signaling connection to an endpoint or other system (such as a Skype for Business / Lync Edge Server) need to be configured with the appropriate certificates to allow that endpoint/system to communicate with Pexip Infinity. If you subsequently deploy more Transcoding Conferencing Nodes to increase conferencing capacity, you do not need to add certificates onto those additional nodes.
- You only need to set up and maintain DNS records for the Proxying Edge Nodes. If you subsequently add more Transcoding Conferencing Nodes you do not have to update any call routing logic on third-party systems that connect to Pexip Infinity.
- When an endpoint has established a signaling and a media connection to one or more Proxying Edge Nodes, the ports used for that connection will not change (even if, for example, the call is transferred to a VMR via a Virtual Reception).
- The servers hosting Proxying Edge Nodes do not require as high a specification as those servers hosting Transcoding Conferencing Nodes. This is because proxying nodes are not as processor intensive as transcoding nodes. The minimum functional CPU instruction set for a proxying node is AVX, which was first available in the Sandy Bridge generation. You still need multiple proxying nodes for resilience and capacity. We recommend allocating 4 vCPU and 4 GB RAM (which must both be dedicated resource) to each Proxying Edge Node, with a maximum of 8 vCPU and 8 GB RAM for large or busy deployments.

For more information, see [Deployment guidelines for Proxying Edge Nodes](#).

Bandwidth optimization

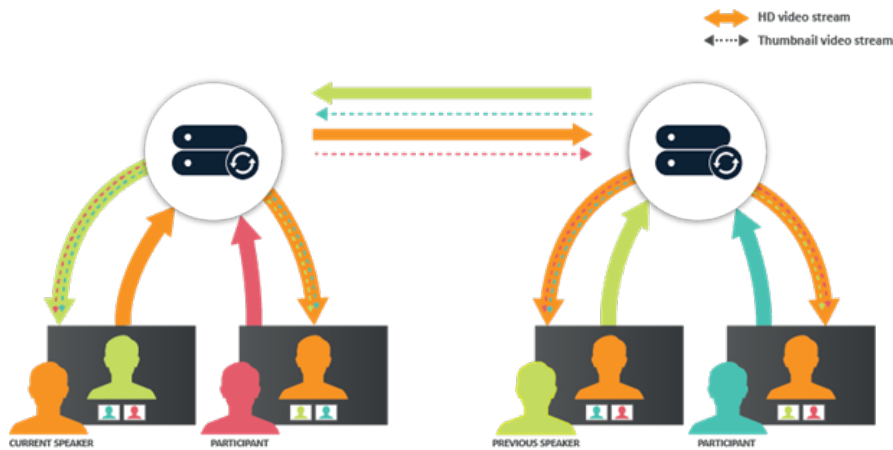
The Pexip Infinity distributed architecture provides major bandwidth savings when compared with traditional MCU deployments.

Bandwidth usage (1 + 7 layout)

When Conferencing Nodes that are hosting a conference (in the default 1 + 7 layout) communicate with each other over a backplane:

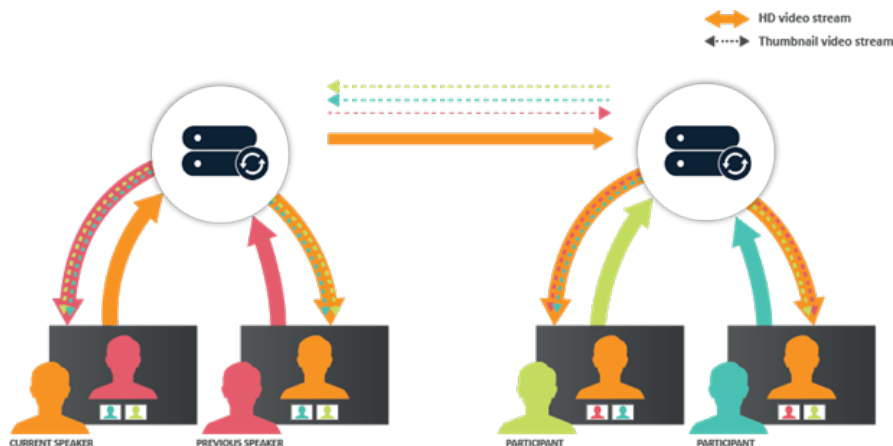
- The Conferencing Node to which the current speaker is connected sends an HD video stream of the current speaker, plus a lower-resolution thumbnail (a smaller image at the bottom of the screen which shows the participant's video) of each of the other participants connected to it.

- The Conferencing Node to which the previous speaker is connected sends an HD video stream of the previous speaker, plus lower-resolution thumbnails of each of the other participants connected to it.
- All other Conferencing Nodes send lower-resolution thumbnails of each of the participants connected to it.



Bandwidth usage when current and previous speakers are connected to different Conferencing Nodes (1 + 7 layout)

As shown in the diagram above, if the current and previous speakers are connected to different Conferencing Nodes, there is one HD stream in each direction in the backplane between the two nodes, plus one smaller stream for every participant being shown in a thumbnail.



Bandwidth usage when current and previous speakers are connected to the same Conferencing Node (1 + 7 layout)

As shown in the diagram above, if the current and previous speakers are connected to the same Conferencing Node, that Conferencing Node sends an HD video stream of the current speaker plus lower-resolution thumbnails of each of the other participants, to all other Conferencing Nodes. The other Conferencing Nodes send lower-resolution thumbnails of the other participants.

This architecture means that there are major bandwidth savings when compared with traditional MCU deployments where all conference participants regardless of location connect to the same MCU and individual HD and thumbnail video streams are sent between the MCU and every endpoint.

Usage guidelines and other conference layouts

The examples above are from the perspective of a conference using a "1 + 7" layout (1 large main speaker and up to 7 other thumbnail participants).

When using other layouts:

- In a 4 + 0 layout, when there are 1 or 2 participants to display, their streams are sent at high resolution over the backplane between nodes; if there are 3 or more participants then up to 5 streams are sent at a medium resolution.
- In a 2 + 21 layout, up to 3 HD streams may be sent between any 2 nodes (up to 2 current speakers and potentially 1 previous speaker).
- When using Adaptive Composition, different resolutions are sent as required for each position in the layout. Up to 13 streams could be sent between nodes: 3 top-row streams (one extra so that current speakers do not see themselves), 3 center-row streams and 7 bottom-row streams, all at their respective call qualities as required for the conference.

You should also note that:

- The bandwidth required by a HD stream on a backplane depends on a combination of many factors. Typically it uses about 1.6 Mbps but could use up to 4 Mbps.
- One thumbnail stream uses about 64-192 kbps.
- A presentation stream uses an additional 1.6 Mbps from the presenter.
- Pexip Infinity supports up to 6 Mbps per participant (this varies depending on the deployment environment, video resolutions, etc).

Downspeeding

Pexip Infinity will automatically downspeed and upspeed individual calls in response to fluctuating network conditions.

Bandwidth restrictions

Administrators can individually restrict the bandwidth available to participants accessing each Virtual Reception, Virtual Meeting Room and Virtual Auditorium, and per Call Routing Rule. Restrictions can also be applied across an entire deployment. Bandwidth limitations cannot be applied to the forwarding connection between a Proxying Edge Node and a Transcoding Conferencing Node. For more information, see [Managing and restricting call bandwidth](#).

Load balancing, redundancy and scalability

This topic summarizes the [load balancing](#), [redundancy](#) and [scalability](#) aspects of Pexip Infinity.

Load balancing

Pexip Infinity load-balances intelligently across all Conferencing Nodes that are grouped within a system location, and can also utilize media overflow locations when all of the conferencing resource within a location has reached its capacity.

Signaling

- The load balancing of signaling across Conferencing Nodes in a single location is achieved via your call management system and/or [DNS SRV records](#). They should be configured so that calls from your endpoints or other systems are routed to only those Conferencing Nodes that you want to receive signaling.
- Signaling always remains routed to the node that received the call. This could be a Proxying Edge Node or a Transcoding Conferencing Node.

Media

- If the signaling is received on a Proxying Edge Node, media proxying is allocated to the proxying node with the most available capacity in the location that received the signaling. The selected proxying node will always handle the media connection with the endpoint, acting as a proxy between the endpoint and a Transcoding Conferencing Node (which should be in a different location).
- If the signaling is received on a Transcoding Conferencing Node, then whichever transcoding node is selected to process the media (which may be a different node to the signaling node) will also directly handle the media connection with the endpoint.
- You can nominate which location's Transcoding Conferencing Nodes to use to process the call media (either directly or proxied), based on the location of the node that is handling the call signaling. If a transcoding node in the nominated location is already processing media for the conference, and it has spare capacity, then it will also process the media from the new caller, otherwise the transcoding node that currently has the most available capacity is selected to process the media (up to a maximum of three transcoding nodes per location per conference instance). If there is no transcoding resource available in that location, then transcoding nodes in the overflow locations (if configured) are used. The selection of the transcoding node in the media overflow location follows the same balancing rules: if a node is currently processing the conference media and has capacity to take the new

call then that node is selected, otherwise the node in the overflow location that currently has the most available capacity is selected.

For full details about how media is routed, load-balanced and directed to overflow locations, see [Handling of media and signaling](#).

For further information about how to configure your specific call management system to work with Pexip Infinity, see the following documentation:

- [Pexip Infinity and Microsoft Skype for Business / Lync Deployment Guide](#)
- [Pexip Infinity and Cisco VCS Deployment Guide](#)
- [Pexip Infinity and Cisco Unified Communications Manager Deployment Guide](#)
- [Pexip Infinity and Polycom DMA Deployment Guide](#)

Redundancy

Management Node

The Management Node is used to manage Conferencing Nodes, configure conference settings, manage licenses and collate system logs. It does not handle any call processing, so if it were to become unavailable the Pexip Infinity service would be unaffected. However:

- You would not be able to make changes to services (Virtual Meeting Rooms, Virtual Receptions and so on).
- Logging would be affected: the Management Node would not receive and collate logs from the Conferencing Nodes. This can be mitigated by using a [syslog server](#) to collate logs.
- After 14 days, licenses would cease to be allocated and the Pexip Infinity service would no longer allow calls.

Conferencing Nodes

You should ensure that your call control system is set up so that, in the event of a Conferencing Node failing, new calls to the conference are routed to another available Conferencing Node.

Basic alternate gatekeeper support is available for H.323 endpoints. When a H.323 device makes a registration request, the Conferencing Node returns a list of any alternate nodes in the same Pexip system location; the device will attempt to use an alternate should the original node be unresponsive.

If multiple Conferencing Nodes are configured in a single location, new calls to a conference can be routed by the call control system to only those nodes that are currently available. To achieve this, you must configure the call control system with all possible Conferencing Nodes (either explicitly or through DNS records), and for liveness checks to be carried out.

Note that there is no live failover; calls in progress on a Conferencing Node that fails will be lost, and participants will need to redial to reconnect to the conference.

Scalability

To increase the capacity of the Pexip Infinity platform, simply deploy one or more new Conferencing Nodes. You may also need to increase the number of [call licenses](#).

Customizing the Pexip Infinity user experience

You can easily apply your own corporate branding to the Pexip Infinity platform, and produce a personalized user experience for all of your Pexip Infinity services.

This topic gives some examples of how [themes](#) can be used to customize the voice prompts and images used in your Pexip Infinity services, and how branding customizations can be applied to change the look and feel of the [Infinity Connect clients](#) (Pexip's free video clients, which can also be used to access those services).

Themes

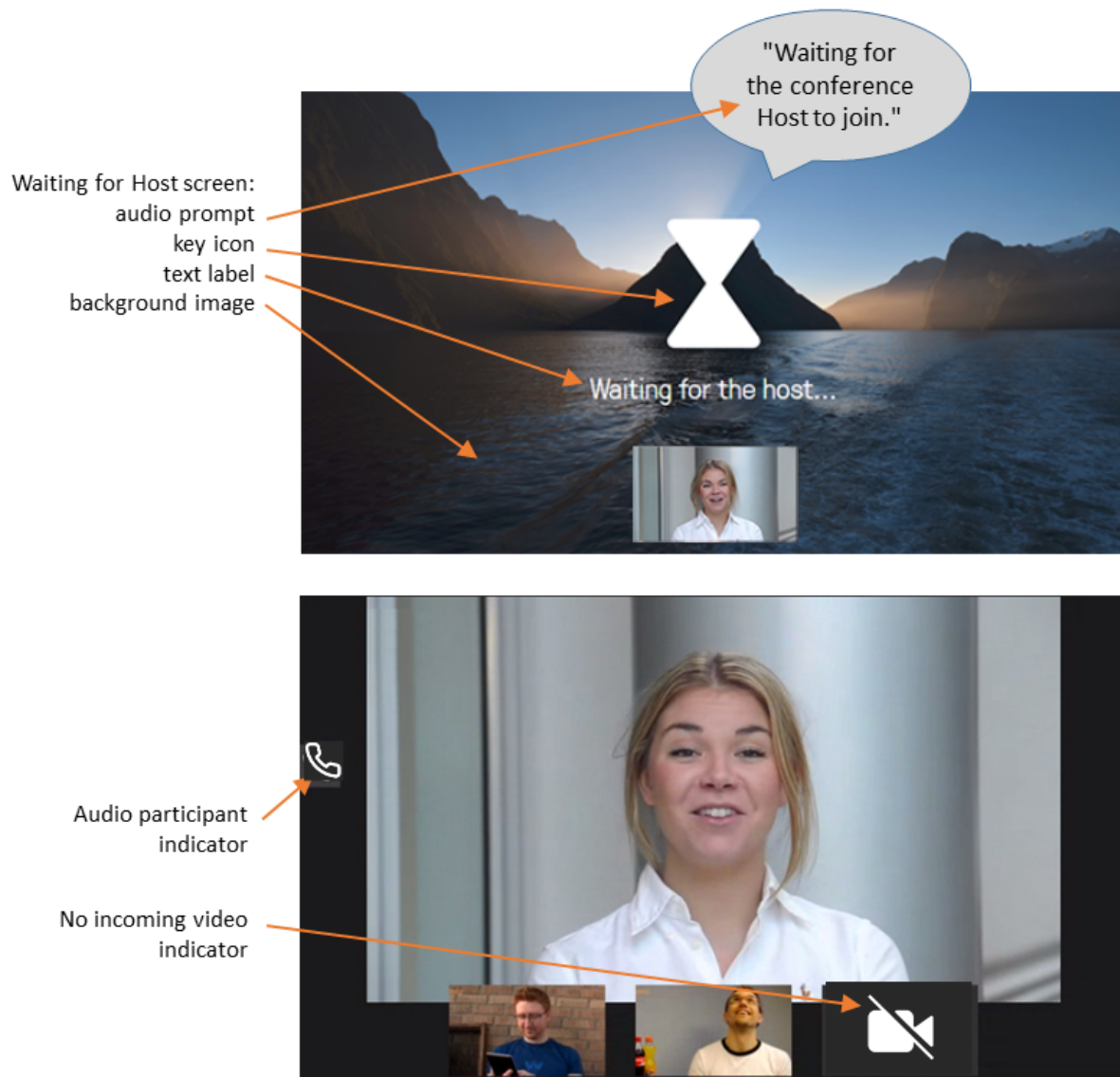
Themes are applied to one or more [services](#) (such as a VMR, Virtual Auditorium, Virtual Reception, or Infinity Gateway call routing rule), and affect all participants using that service, regardless of the device or video client that the participant is using to make the call. A different theme can be applied to each service if required.

Themes are used to control:

- all of the voice prompts used when accessing or participating in a conference
- the appearance of PIN entry screens (on devices other than the Infinity Connect clients)
- in-conference indicators such as audio-only participants, streaming, no incoming video etc. and the watermark
- the appearance of information screens such as waiting for host, PIN entry, capacity exceeded etc.

The following images show some examples of which elements of the user experience can be customized via themes. All of the examples show a video client that is accessing a service that uses the base (default) Pexip theme.





Themes are uploaded and managed via the Pexip Infinity Administrator interface (**Services > Themes**).

For more information, including the full set of theme elements that can be customized, see [Customizing conference images and voice prompts using themes](#).

Infinity Connect customization

The branding and styling of the Infinity Connect clients (web app and desktop) can be customized. This changes the look and feel of the Infinity Connect client regardless of which service is being accessed. (However, the theme-based elements of each individual service may also have been customized — a theme changes the look and feel of the actual conference you have joined, or are trying to join.)

Infinity Connect customization can be used to control:

- default settings such as bandwidth, screen sharing frame rate and so on
- the ability to display an image/logo and accompanying welcome text on a landing page, and to use a custom favicon
- language translations and the default language
- the color scheme for buttons, icons and other graphic indicators; elements can be customized individually or a general color scheme can be applied to all similar items.

To customize the web app you typically create and then upload a branding package to the Management Node. That branding package is then automatically applied to all users of the web app. To apply the same customized branding to the desktop clients you need to use Pexip Infinity's provisioning features to instruct those clients to override their built-in branding and use the customized branding instead.

The recommended method to create a branding package for the Infinity Connect clients is to use the Pexip branding portal (<https://brandingportal.pexip.com>).

A beginner's guide to Pexip Infinity

If you're new to video conferencing, or just to Pexip Infinity, this section covers the basics of what you need to know about the Pexip Infinity solution.

First, a bit of history

How it used to be

In the past, the use of video conferencing within businesses was usually restricted to a special AV unit installed in a dedicated meeting room, which had to be booked in advance. Companies had to install special servers called Multipoint Control Units (MCUs) to run their videoconferencing, and because these units were big and expensive, they would be installed in just a few central locations. Video also used up a lot of expensive bandwidth. This all meant that the use of videoconferencing was restricted.

A new way of working

Now that the use of video has become more pervasive at home and in business, and computing resources are faster and cheaper, users have come to expect instant access to video calls from their desktop. The Pexip Infinity solution enables organizations to provide universal access to videoconferencing. It replaces the old dedicated hardware MCUs with software that can be installed on standard servers and run as virtual machines, and its [distributed architecture](#) is designed to allow as many of these virtual machines as required to be spread around a range of locations, providing videoconferencing resources where and when it is required, reducing [bandwidth usage](#) across the organization.

What's wrong with using what I have now?

There are plenty of free video calling and videoconferencing solutions out there, and your organization might already be using some of them. But these systems aren't secure, and they are limited in the number of people who can connect into a single meeting. They also don't support connections from others not using the same solution, so you often can't connect with others outside your organization, or who want to call using a device such as a telephone.

What to look for in a modern videoconferencing solution

Lets any device talk to any other

Most organizations have a mix of endpoints that are used to make video calls. Meeting participants might want to call from a dedicated meeting room equipped with a big-screen video endpoint; a desktop client such as Skype for Business; a web browser; or even a telephone. In the past, these systems weren't able to connect to each other. But Pexip Infinity can act as an interpreter for all these systems, allowing participants to use whatever device they prefer to call in to a meeting. We even offer our own [desktop, mobile and web-based clients](#) for those users who don't have access to traditional video devices.

Virtual meeting rooms for everyone

[Pexip Infinity Virtual Meeting Rooms](#) (VMRs) are always available and ready to be used. VMRs that aren't currently being used don't take up any resources, so you can create one for every person in your organization. Whenever someone wants to hold a meeting over video, they can just invite the participants into their VMR.

Making direct calls to other people or other meeting platforms

Pexip Infinity doesn't just provide VMRs – the Infinity Gateway feature also allows users to make direct calls to each other. This is particularly significant if the two people on the call are using different video devices or solutions that would not otherwise be able to connect to each other.

It also lets you join meetings that are running on other systems such as Microsoft Teams or Google Meet.

Registering your device so it can be called

If you have access to a video endpoint, you can usually use it to make outbound calls. However, if you want to be able to receive calls, you need to register your device and the address that can be used to find it, so that others can make calls to it. There are special

systems that can handle registrations within an organization, but Pexip Infinity has this functionality built in.

The advantages of software

Latest features

Because Pexip Infinity is simply software, you have immediate access to the latest releases and all the new features – for free. Just download the software and install it on your servers – it really is that simple.

Add and remove resources as required

As long as you have access to computing resources – either on-premises servers or cloud-based solutions such as AWS or Azure – you can create additional instances of Pexip Infinity to provide any additional capacity, as and when required, and in a matter of minutes. This can even happen [automatically](#).

Next steps

- [Choosing a deployment environment](#) explains your options for deploying Pexip Infinity on your own servers or on a cloud service.
- Our [installation guides](#) then provide full information about how to obtain and install the Pexip Infinity software on your chosen platform.

Using your Virtual Meeting Room

Pexip Infinity provides any number of users with their own personal Virtual Meeting Room (VMR) which they can use to hold conferences, share presentations, and chat.

Here's what you need to know for you and your guests to use and make the most of your VMR.

It's always available

Your VMR is a personal space that is **always available** for you to meet with others over video or audio and share content. You can use it whenever you want — for pre-arranged meetings or for ad hoc calls — without making a reservation in advance.

VMR addresses and PINs

Your administrator will let you know the address of your VMR. Often you will have two addresses — one will typically be in the format `name@example.com` (like an email address), and the other will be a number, for example `555678`. Which address you and your guests will use depends on the type of video device they are using.

In most cases you will be able to dial your VMR directly using the email-style address, but in some circumstances you may need to go through a central Virtual Reception, and then enter your VMR number.

Your VMR may be set up with a PIN code that you as the host can use, which will enable you to control the conference (for example, locking it, and muting guests). Your guests may be able to join your VMR without a PIN, or they may need to enter a different Guest PIN.

Use any device

You and your guests can use almost any video-enabled device — or even a normal telephone — to join a meeting in your VMR, including:

- **A video conferencing endpoint:** on most video endpoints and room systems you can simply enter the address of your VMR.
- **A client like [Skype for Business](#) / [Lync](#) or [Cisco Jabber](#):** again, you should just be able to enter the address of your VMR.
- **One of the Pexip [Infinity Connect](#) clients:** these are available for [desktop](#) and [mobile](#) (iOS and Android). If you use one of the Infinity Connect clients, you also get built-in access to a wide selection of meeting information and controls, including a list of attendees. Again, simply enter the VMR address.
- **A [web browser](#):** if you or your guests don't have a fancy endpoint or a software client, you can still call into your VMR from almost any browser including Chrome and Edge — no downloads required! Your administrator will need to tell you the link to use in your browser, and then from there you can enter your VMR address to join the meeting.
- **A telephone:** if you want to join the conference over audio, then your administrator will tell you the number you can dial.

Using your keypad to control the conference

If you aren't using Infinity Connect, you can still control some aspects of the conference from your endpoint's keypad (using what are known as DTMF controls).

The default DTMF controls that can be used within a conference are:

DTMF digits	Control
*7	Toggle conference lock and unlock
*5	Toggle mute and unmute all Guests
##	Terminate the conference (disconnect all participants including yourself)

Try out the features in Infinity Connect!

There are many features built in to the Pexip Infinity Connect clients, which are available for desktop, mobile and browsers. Most are self-explanatory, but here are some of the most popular (see the relevant [user guide](#) for a full list):

- **Controlling the meeting room:** when you've joined the meeting room, you can lock it and then only let in the people you want. You can also dial out to people that you want to bring into the meeting.
- **Viewing a list of participants:** you'll see a list of conference attendees on the left of your screen. From here you can mute and disconnect any of them, or change their role from Guest to Host.
- **Sharing content:** you and your Guests can share PDFs and images with others in your VMR. Depending on your client, you may also be able to share your screen.
- **Sending messages:** at the bottom left of the **Events** tab (in the side panel) is a text box. Type messages here to send them to other participants who have joined your conference using either an Infinity Connect or Skype for Business / Lync client. You can also share URLs, and they will appear to other participants as clickable links.

Connecting with Skype for Business / Lync

Pexip VMRs fully support Skype for Business / Lync, so you and your colleagues can keep using the clients you are familiar with to connect to your VMR. Below are just some of the features we support (for full details, see [Using Skype for Business / Lync clients](#)):

- **Adding your VMR as a contact:** you can add your VMR as a Sfb/Lync contact just like you would anyone else. Simply type the address of your VMR into the search box – it should appear in your contacts list, and you can then right-click it to add as a favorite.
- **Sharing and viewing content:** you can share your screen, program, or PowerPoint presentation with others in your VMR just as you would in a normal Sfb/Lync call. And when others in the VMR share their content, you'll see it just as expected.

Pexip Infinity installation guidelines

Pexip Infinity is installed as a Management Node and one or more Conferencing Nodes. All nodes are virtual machines (VMs) that are either deployed onto host hardware using hypervisors, or hosted on a cloud platform service. The Management Node is deployed first, and is then used to configure the Pexip Infinity platform and deploy Conferencing Nodes.

For information about how to install and deploy Pexip Infinity, see:

Installation overview	45
Planning and prerequisites	45
Testing and next steps after initial installation	90

To upgrade your existing Pexip Infinity platform to the latest software version, see [Upgrading the Pexip Infinity platform](#).

Installation overview

Pexip Infinity is installed as a Management Node and one or more Conferencing Nodes. All nodes are virtual machines (VMs) that are either deployed onto host hardware using hypervisors, or hosted on a cloud platform service. The Management Node is deployed first, and is then used to configure the Pexip Infinity platform and deploy Conferencing Nodes.

Full instructions on how to install Pexip Infinity, including how to install the Management Node, are available [on our website](#) and to download as individual hypervisor-specific [installation guide PDFs](#).

To upgrade your existing Pexip Infinity platform to the latest software version, see [Upgrading the Pexip Infinity platform](#).

Planning and prerequisites

Choosing a deployment environment

Pexip Infinity supports a wide range of network scenarios, from small, local, private deployments to large, global, cloud-based implementations. How many Conferencing Nodes you deploy, where they are located, and the hypervisors you use will depend on the makeup and requirements of your individual organization.

While your Pexip authorized support representative is the best person to advise you on the most appropriate deployment environment, below are some of the factors to consider.

On-premises, cloud or both?

On-premises deployments use your own network, hardware and hypervisors. You will need to source appropriate servers to host the Management Node and Conferencing Nodes (for more information see [Server design recommendations](#)). You will also need to consider whether the majority of users are connecting from within your private network or whether you will have users connecting externally, and therefore if your Conferencing Nodes need to be publicly accessible. If you already have a mature IT network with datacenters in appropriate locations, this may be a suitable option.

Cloud deployments use Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP) or Oracle Cloud Infrastructure to host some or all of your Pexip nodes. This option does not require initial investment in hardware; instead you must have an agreement with the cloud service provider, and you pay for usage.

You can deploy Pexip Infinity using a **combination of on-premises and cloud**, with the Management Node and some Conferencing Nodes deployed within your own network, and other Conferencing Nodes deployed using a cloud service and/or in the Pexip Private Cloud, for increased flexibility. You can even set up Dynamic bursting to a cloud service to give you temporary increases in capacity whenever you need it.

There is also the option to use the **Pexip Private Cloud**, where Pexip deploys some or all of your Transcoding Conferencing Nodes on your behalf in the form of [Pexip Smart Scale locations](#). This option is useful if you want the privacy and security advantages of a private cloud deployment without having to set it up yourself.

You don't have to stay committed to one particular deployment environment, and you can always increase or decrease the number of Conferencing Nodes as your capacity requirements evolve. You could, for example, start with an on-premises deployment and then add some cloud-hosted nodes later.

For more details on the on-premises and cloud options, see [Network deployment options](#).

Hypervisor

For on-premises deployments, you need to install a hypervisor on your servers before you can deploy the Management Node and Conferencing Nodes. Pexip Infinity currently supports four hypervisors:

- **VMware** is an independent product with a range of features, dependent on the licensing options you choose.
- **Microsoft Hyper-V** is included with Windows Server and also offers a range of features.
- **KVM** and **Xen** are open-source options that are not as feature-rich.

For more information, see [Supported hypervisors](#).

Number, location and role of Conferencing Nodes

The capacity of your Pexip Infinity deployment, in terms of how many calls can be handled at any one time, is largely determined by the number of Conferencing Nodes you deploy. To optimize the conferencing experience, we recommend that, where possible, you locate Conferencing Nodes close to each of your main user bases. For example, if the majority of your users are located in three main offices then you should consider concentrating the majority of your Conferencing Nodes among those three locations. If however the majority of your users are spread across many geographic locations, you should deploy Conferencing Nodes evenly among your datacenters.

You must also consider the role of your Conferencing Nodes, in terms of which nodes will be accessed by the client devices and endpoints that need to participate in conferences. In a large Pexip deployment with 5 or more Conferencing Nodes, or where you need to transcode media in multiple locations such as within a DMZ, you should consider deploying Proxying Edge Nodes in addition to your Transcoding Conferencing Nodes.

For more information, see [Capacity planning](#) and [Distributed Proxying Edge Nodes](#).

Server and network requirements

This topic summarizes the host server and network requirements when deploying Pexip Infinity.

Host servers

The Management Node and Conferencing Nodes are virtual machines (VMs) that run on industry-standard host servers. A Management Node can run on the same host server as a Conferencing Node. Other Conferencing Nodes can run on host servers in the same or different locations, allowing you to create a globally distributed system.

Our [Pexip Infinity Server Design Guide](#) help you choose and configure appropriate servers on which to host your Management Node and Conferencing Nodes.

Network requirements

Depending on your network deployment scenario, you may have to configure your system to operate behind a static NAT and to ensure that traffic can be routed between nodes. See [Network deployment options](#) and [Network routing and addressing options for Conferencing Nodes](#) for more information about the requirements and implications of your deployment scenario.

Note that in all Pexip Infinity deployment scenarios:


- The Management Node must be able to reach all Conferencing Nodes (Proxying Edge Nodes and Transcoding Conferencing Nodes) and vice versa.
- Each Conferencing Node must be able to reach every other Conferencing Node (Proxying Edge Nodes and Transcoding Conferencing Nodes), except:
 - When a location contains Proxying Edge Nodes, those nodes only require IPsec connectivity with:
 - any other proxying nodes in that location
 - all nodes in the transcoding location, and the primary and secondary overflow locations that are associated with that location
 - the Management Node.

This means that the proxying nodes in one location do not need to have a direct network connection to other proxying nodes in other locations.

- Any internal firewalls must be configured to allow UDP port 500 and traffic using IP protocol 50 (ESP) in both directions between all Pexip nodes.
- There cannot be a NAT between any Pexip nodes.

NTP servers

Pexip Infinity uses NTP servers to obtain accurate system time. This is necessary to ensure correct operation, including configuration replication and log timestamps.

-  All host servers **must** be synchronized with accurate time before you install the Management Node or Conferencing Nodes on them.

- i** NTP **must** be enabled on the Management Node VM before you deploy any Conferencing Nodes (this is done during installation of the Management Node).

We strongly recommend that you configure at least three distinct NTP servers or NTP server pools on all your host servers and the Management Node itself. This ensures that log entries from all nodes are properly synchronized.

The VMs hosting the Management Node and Conferencing Nodes use the UTC timezone, and all logs are in UTC. Do not attempt to change the timezone on these systems. Note however that the administrator web interface uses your local time.

DNS servers

Pexip Infinity uses DNS to resolve the hostnames of external system components including NTP servers, syslog servers, SNMP servers and web proxies. It is also used for call routing purposes — SIP proxies, gatekeepers, external call control and conferencing systems and so on. The address of at least one DNS server must be added to your system.

After configuring the DNS servers available to your system, you should assign appropriate DNS servers to each location (**Platform > Locations**). Each Conferencing Node in that location will then use those DNS servers.

Using a VPN to connect to Pexip Infinity

Pexip Infinity is designed to remove the need for Virtual Private Networks (VPNs). We recommend that you deploy Proxying Edge Nodes to enable external connectivity to your enterprise deployment, instead of asking users to connect to internally-located nodes over a VPN. In most cases, connections to Pexip Infinity over a VPN work successfully, but in some cases a lower MTU may be required, or it could lead to UDP media being TCP encapsulated which can cause additional latency.

Call control

If your deployment includes a call control system, it must be configured to route calls to Pexip Infinity appropriately. The exact configuration will depend on your deployment and dial plan, but in general calls placed from an endpoint to a Virtual Meeting Room alias should be routed to the endpoint's local Conferencing Nodes. For more information, see [Implementing a dial plan](#).

Skype for Business / Lync server

If your deployment includes on-premises Skype for Business / Lync, you must set up static routes to domains used by Pexip Infinity aliases. For more information, see [Pexip Infinity and Microsoft Skype for Business / Lync Deployment Guide](#).

Capacity planning

The capacity of each Transcoding Conferencing Node in your Pexip Infinity deployment — in terms of the number of connections* that can be handled simultaneously — depends on a variety of factors including:

- The [server capacity](#) and hardware configuration.
- The [type of call](#) — Full HD, HD, SD, or audio-only, the codec, and whether there is a presentation stream included.
- The [number of unique VMRs](#) being used (and thus the number of backplanes being reserved).
- The [type of gateway call](#) — whether the inbound and outbound legs are on the same transcoding node or not, and the types of client involved in the call.

* A connection can be a call or presentation from an endpoint to a Virtual Meeting Room or Virtual Auditorium, a backplane between Transcoding Conferencing Nodes, or a call into or out of the Infinity Gateway. In this context, a **connection** is analogous to a **port**. In some situations, a single conference participant such as a WebRTC or Skype for Business client requires two connections (one for the video call, and one for presentation content).

When a connection is proxied via a Proxying Edge Node, the proxying node also consumes connection resources in order to forward the media streams on to a Transcoding Conferencing Node. A transcoding node always consumes the same amount of connection resources regardless of whether it has a direct connection to an endpoint, or it is receiving the media streams via a proxying node.

In all cases, you must also have sufficient concurrent [call licenses](#) available.

The following sections explain each of these factors. For some comprehensive examples showing how these different factors can combine to influence capacity, see [Resource allocation examples](#).

Server capacity

The capacity and configuration of the server on which the Conferencing Node is running determines the number of calls that can be handled. This is influenced by a number of factors, including processor generation, number of cores, processor speed, hypervisor and BIOS settings.

Call types and resource requirements

The type of call (Full HD, HD, SD, or audio-only) affects the amount of resource required by a Transcoding Conferencing Node to handle the call.

In general, when compared to a single high definition **HD** 720p call:

- a **Full HD** 1080p call uses twice the resource
- an **SD** standard definition call uses half the resource
- an **audio-only** call uses one twelfth of the resource.

A WebRTC call using the **VP8** codec uses the same amount of resource as H.264, and the **VP9** codec uses around 25% more resource, so VP9 at 720p uses the equivalent of 1.25 HD resources, and VP9 at 1080p uses the equivalent of 2.5 HD resources. WebRTC clients also use 0.5 HD additional resources for sending presentation content and 1 additional HD resource when receiving full motion presentation. Note that within the same conference some participants may use VP9 (if they are connected to a Conferencing Node using the AVX2 or later instruction set) while other participants may use VP8 (if they are connected to a Conferencing Node on older hardware).

Conferences that use the **Adaptive Composition** layout consume additional Conferencing Node resources. The actual amount of additional resource depends on many factors, but as a guide, it uses an additional 0.5 HD of resource for each video participant that is on stage (up to 13 participants per conference). This is regardless of the call quality / resolution of the conference itself and each individual participant's connection (codec, bandwidth and so on).

If you want to limit video calls to specific resolutions (and limit the transcoding node resources that are reserved for calls), you should use the **Maximum call quality** setting.

On startup, each Conferencing Node runs an internal capacity check. This capacity is translated into an estimated maximum number of Full HD, HD, SD or audio-only calls, and can be viewed on the status page (**Status > Conferencing Nodes**) for each Conferencing Node. The status also shows the current media load on each Conferencing Node as a percentage of its total capacity.

Proxying Edge Node resource requirements

When a connection is proxied via a Proxying Edge Node, the proxying node also consumes connection resources in order to forward the media streams on to a Transcoding Conferencing Node.

A proxying node uses approximately the equivalent of 3 audio-only resources to proxy a video call (of any resolution), and 1 audio-only resource to proxy an audio call.

We recommend allocating 4 vCPU and 4 GB RAM (which must both be dedicated resource) to each Proxying Edge Node, with a maximum of 8 vCPU and 8 GB RAM for large or busy deployments.

Backplane reservation

Each conference instance on each Transcoding Conferencing Node reserves a backplane connection at a resource level corresponding to the conference's **Maximum call quality** setting, to allow the conference to become geographically distributed if required. The exceptions to this are:

- Deployments with a single Conferencing Node. In such cases, no backplanes will ever be required, so capacity is not reserved.
- Conferences that are audio-only (in other words, where the conference has its **Conference capabilities** set to **Audio-only**). In such cases, capacity equivalent to one audio connection is reserved for the backplane.

For some reservation examples, see [Resource allocation examples](#).

Gateway calls

Gateway calls (person-to-person calls, or calls to an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet) require sufficient capacity for both the inbound leg and the outbound leg. In general, this means that each gateway call consumes resources equivalent to two connections.

Non-distributed gateway calls (where the inbound and outbound legs are on the same transcoding node) involving only SIP or H.323 clients do not use any additional ports/connections. However, in other scenarios, additional ports may be used as described below (assuming that **Maximum call quality** is HD unless otherwise specified):

- Distributed gateway calls (where the outbound leg is on a different transcoding node to the inbound leg) consume backplane ports — thus 1 HD video + 1 backplane for participant A plus 1 HD video + 1 backplane for participant B. This typically occurs when calling registered endpoints (where the outbound call to the registered endpoint will originate from the node the endpoint is registered to), or when using Call Routing Rules with an **Outgoing location** set to something other than **Automatic**.
- For non-distributed gateway calls involving a Skype for Business / Lync client, a backplane is reserved in case the SfB/Lync user starts presenting (as the RDP/VbSS presentation stream could connect to any Conferencing Node due to DNS). Each presentation stream counts as 1 HD port. Thus if the incoming RDP/VbSS call lands on the same node as the video call then the resource usage is equivalent to 3 HD ports (2 video + 1 presentation). If the RDP/VbSS call lands on a different node to the video call then the resource usage for the call is equivalent to 5 HD ports (2 video + 2 backplane + 1 presentation).
- For a gateway call to a **Microsoft Teams** meeting, the connection to Teams uses 1.5 HD of resource if **Maximum call quality** is SD or HD, otherwise it uses 1.5 Full HD resources. The resources required for the VTC leg of the connection depend upon the **Maximum call quality** setting. No additional resources are required for the connection from Pexip Infinity to Teams for presentations to or from the Teams meeting.
- For a gateway call to a **Skype for Business meeting**, the connection to SfB uses 1 HD of resource for main video and will use another 1 HD of resource if either side starts presenting. The resources required for the VTC leg of the connection depend upon the **Maximum call quality** setting.
- For a gateway call to **Google Meet**, the connection to Google Meet always uses 1 HD resource (it uses VP8) for main video. The resources required for the VTC leg of the connection depend upon the type of endpoint and the **Maximum call quality** setting. If the VTC endpoint starts to present content then an extra 1 HD resource is used for the connection from Pexip Infinity to Google Meet. However, no additional resources are required if presentation content is sent from Google Meet.

Call protocols and presentation content

The various call protocols have different behavior and limitations for sending and receiving video and presentation content, in addition to what Pexip Infinity will request based on its configured maximum call quality and bandwidth settings. As the endpoints ultimately decide what to send to Pexip Infinity, the following information should be seen as a guide only.

WebRTC

WebRTC may send resolutions up to 1080p to Pexip Infinity, depending on the camera capabilities and available bandwidth. Presentation content is always sent in a separate channel to main video, at up to the same resolution as the main video channel.

When Infinity Connect WebRTC clients elect to receive presentations in full motion, they are received in a separate channel with the same bandwidth as main video. Standard presentations are received as JPEG content in a separate channel with no fixed bandwidth. The JPEG content will update every second or so, and the amount of bandwidth depends on the amount of change from the previous JPEG image.

Skype for Business / Lync

Skype for Business / Lync clients may send resolutions up to 1080p to Pexip Infinity, depending on the camera capabilities and available bandwidth. RDP and VbSS presentation content is sent in a separate channel to main video. RDP is more bandwidth heavy than VbSS and it utilizes TCP, which is a poor choice for realtime media. RDP content is sent at the same resolution as the originating screen.

Skype for Business / Lync clients ask to receive content at a resolution based on the size of the window they are appearing in. For example, at smaller window sizes, the client will request video to be sent to it from Pexip Infinity at CIF (352x288) but as the size of the window expands, this can increase up to 1080p.

H.323 and SIP

H.323 and SIP video endpoints may send resolutions up to 1080p to Pexip Infinity at whatever bandwidth and frame rate they decide. Presentation content shares the bandwidth of the main video channel, meaning that when a participant starts presenting, this might decrease resolution of the main video. Most H.323 and SIP endpoints prioritize motion (higher frame rate) for main video and sharpness (higher resolution) for content.

RTMP

RTMP clients send resolutions up to 720p to Pexip depending on camera and bandwidth. RTMP does not support 1080p. RTMP clients do not support sending content, but they do receive content as JPEG.

Licenses

The total number of concurrent calls that can be made in your deployment (regardless of whether those calls are HD, SD or audio-only) is limited by your license. For more information, see [Pexip Infinity license installation and usage](#).

Scaling up capacity, media overflow and dynamic bursting

You can easily scale a deployment up by creating several Conferencing Nodes in the same location (i.e. the same datacenter). Capacity can even be added “on the fly” – Conferencing Nodes can be added in a couple of minutes if more capacity is needed. Alternatively, each location can be configured to overflow to another location if it reaches its capacity, including bursting to temporary resources on a cloud service.

For more information on media overflow and dynamic bursting, see:

- [Handling of media and signaling](#)
- [Dynamic bursting to a cloud service](#)

Supported hypervisors

Pexip Infinity currently offers specific support for VMware ESXi, Microsoft Hyper-V, KVM and Xen hypervisors. Other hypervisors and orchestration layers, such as Citrix XenServer, may be used but are not officially supported by Pexip.

Note that suspending and resuming Virtual Machines that are running the Pexip Infinity Management Node or Conferencing Nodes is not supported.

The Pexip Infinity platform can also be deployed as a [cloud service](#) via **Amazon Web Services (AWS)**, **Microsoft Azure**, **Google Cloud Platform**, or **Oracle Cloud Infrastructure**, with private, public or hybrid deployment options.

VMware vSphere ESXi

VMware versions

Version 26 of the Pexip Infinity platform supports VMware vSphere ESXi 6.x.

You must have a suitable VMware environment already installed.

VMware editions

The Pexip Infinity platform will run on the **free edition** of vSphere Hypervisor. However, this edition has a number of limitations (limited support from VMware, no access to vCenter or vMotion). For this reason we do not recommend its use except in smaller deployments, or test or demo environments.

The minimum edition of VMware that we recommend is the vSphere **Standard edition**. This does not have the limitations of the free edition. If you do not already use VMware in your enterprise, the vSphere **Essentials Kit** is a simple way to get started and will provide you with Standard edition licenses for 3 servers (with 2 CPUs each) plus a vCenter license.

The **Enterprise Plus edition** includes further additional features relevant to the Pexip Infinity platform that could be of benefit to larger deployments. These include Storage DRS and Distributed Switch.

For a comparison of the VMware editions, see <http://www.vmware.com/products/vsphere.html#compare>.

Microsoft Hyper-V

Version 26 of Pexip Infinity supports Microsoft Hyper-V in the form of Microsoft Hyper-V Server 2012 and later, or Windows Server 2012 and later. You must have a suitable Hyper-V environment already installed.

KVM

For information on configuring a KVM environment specifically for use with Pexip Infinity, see [Configuring KVM for Pexip Infinity](#).

Xen

Pexip Infinity requires Xen 4.2 and later. For information on configuring a Xen environment specifically for use with Pexip Infinity, see [Configuring Xen for Pexip Infinity](#).

Other hypervisors and orchestration layers

Pexip Infinity Management Nodes and Conferencing Nodes support static IP addressing only. Therefore the hypervisor or orchestration layer must also support static IP addressing.

Cloud service providers

The Pexip Infinity platform can also be deployed as a [cloud service](#) via **Amazon Web Services (AWS)**, **Microsoft Azure**, **Google Cloud Platform**, or **Oracle Cloud Infrastructure**, with private, public or hybrid deployment options.

About the Pexip Infinity software files

The Pexip Infinity platform is entirely software-based, and is deployed by installing the files provided by Pexip onto your hypervisor or cloud platform to create Virtual Machines (VMs), which you then configure appropriately for your environment.

The Pexip Infinity version 26 release includes the files described below. Which set of files you require depends on:

- the hypervisor or cloud platform you are deploying on
- whether you are deploying a new instance of Pexip Infinity, upgrading an existing deployment, or deploying a new Conferencing Node
- if you are deploying new generic Conferencing Nodes in unsupported or non-standard environments.

i **No changes** should be made to any Pexip VM via the terminal interface (other than as described when running the initial Pexip installation wizard) unless directed to do so by Pexip support. This includes (but is not limited to) changes to the time zone, changes to IP tables, configuration of Ethernet interfaces, or the installation of any third-party code/applications.

Upgrading an existing deployment (all environments)

The Pexip Infinity upgrade package is required when upgrading an existing deployment. It is used to upgrade all deployment environments.

Description	File name	Location	Hypervisor	Notes
Pexip Infinity upgrade package	Pexip_Infinity_v26_UPGRADE_<build>.tar	Pexip download page	Any hypervisor or cloud platform	Used when upgrading the Pexip Infinity platform to version 26. For more information, see Upgrading the Pexip Infinity platform .

Deploying the Management Node on an on-prem hypervisor

Use one of these files when deploying the Management Node on an on-prem hypervisor.

Description	File name	Location	Hypervisor	Notes
Pexip Infinity OVA	Pexip_Infinity_v26_generic_pxMgr_<build>.ova	Pexip download page	Any hypervisor except Microsoft Hyper-V	Used to deploy a new generic Management Node VM using VMware, KVM, Xen or other hypervisors except Microsoft Hyper-V.
Pexip Infinity ZIP	Pexip_Infinity_v26_HyperV_pxMgr_<build>.zip	Pexip download page	Microsoft Hyper-V	Used to deploy a generic Management Node VM using Hyper-V.

Deploying new Conferencing Nodes on supported hypervisors

There is no requirement to download any files from the Pexip website when deploying new Conferencing Nodes on VMware, Hyper-V, KVM or Xen hypervisors, as the relevant file is generated on request by the Management Node.

- For VMware, KVM and Xen, a file in the format `pexip-<hostname>.<domain>.ova` is generated. See [Deploying new Conferencing Nodes](#) for full information.
- For Hyper-V, a file in the format `pexip-<hostname>.<domain>.zip` is generated. See [Deploying a Conferencing Node](#) for full information.

Deploying the Management Node or Conferencing Nodes on cloud environments (Azure, AWS, GCP or Oracle)

There is no requirement to download any files from the Pexip website when you are deploying the Pexip Infinity Management Node or Conferencing Nodes on an AWS or Azure cloud platform — the relevant Pexip image files are hosted on the AWS/Azure platform instead. However, GCP and Oracle image files are downloadable from the Pexip website.

For specific deployment information about each platform, see:

- [Deploying Pexip Infinity on Amazon Web Services](#)
- [Deploying Pexip Infinity on Microsoft Azure](#)
- [Deploying Pexip Infinity on Google Cloud Platform](#)
- [Deploying Pexip Infinity on Oracle Cloud Infrastructure](#)

Deploying new generic Conferencing Nodes on non-supported hypervisors or orchestration layers

Use one of these files when deploying a Conferencing Node on other non-supported hypervisors or orchestration layers.

Description	File name	Location	Hypervisor	Notes
Generic Pexip Infinity Conferencing Node OVA	Pexip_Infinity_v26_generic_ConfNode_<build>.ova	https://dl.pexip.com/infinity/index.html and then select the appropriate directory for your software version	Other environments that take .ova or .ovf files as input	Used to deploy a generic Conferencing Node VM in environments that take .ova or .ovf files as input. For more information, see Deploying a Conferencing Node using a generic VM template and configuration file .
Generic Pexip Infinity Conferencing Node ZIP	Pexip_Infinity_v26_HyperV_ConfNode_<build>.zip	https://dl.pexip.com/infinity/index.html and then select the appropriate directory for your software version	Hyper-V in a cloud-based environment	Used to deploy a generic Conferencing Node VM using Hyper-V in cloud-based environments or other orchestration layers where standard deployment is problematic. For more information, see Deploying a Conferencing Node using a generic VM template and configuration file .

Network deployment options

If you need to support business-to-business video calls and provide access to Pexip Infinity resources from external systems and endpoints such as remote or federated Skype for Business / Lync clients, remote SIP and H.323 endpoints, and Infinity Connect clients, you need to consider how to deploy your Pexip Infinity Conferencing Nodes.

This section explains how the Pexip Infinity platform fits into typical network deployment scenarios:

- A **private** "on-premises" deployment of privately-addressed Transcoding Conferencing Nodes, with external connections routed via Proxying Edge Nodes. There are several variations to this option:
 - [Proxying Edge Nodes in combination with third-party video call control](#)
 - [Routing all external calls via Proxying Edge Nodes](#)
 - [Combining with on-premises Microsoft Skype for Business / Lync](#)
- A **public** DMZ deployment with publicly-addressable Pexip Infinity nodes (with optional static NAT).

For additional flexibility you can deploy Conferencing Nodes with [dual network interfaces](#) (one "internal" interface for inter-node communication, and one "external" interface for signaling and media to endpoints and other video devices).

The Pexip Infinity platform can also be deployed as a [cloud service](#) via **Amazon Web Services (AWS)**, **Microsoft Azure**, **Google Cloud Platform**, or **Oracle Cloud Infrastructure**, with private, public or hybrid deployment options.

General network requirements

Note that in all Pexip Infinity deployment scenarios:

- The Management Node must be able to reach all Conferencing Nodes (Proxying Edge Nodes and Transcoding Conferencing Nodes) and vice versa.
- Each Conferencing Node must be able to reach every other Conferencing Node (Proxying Edge Nodes and Transcoding Conferencing Nodes), except:
 - When a location contains Proxying Edge Nodes, those nodes only require IPsec connectivity with:
 - any other proxying nodes in that location
 - all nodes in the transcoding location, and the primary and secondary overflow locations that are associated with that location
 - the Management Node.

This means that the proxying nodes in one location do not need to have a direct network connection to other proxying nodes in other locations.

- Any internal firewalls must be configured to allow UDP port 500 and traffic using IP protocol 50 (ESP) in both directions between all Pexip nodes.
- There cannot be a NAT between any Pexip nodes.

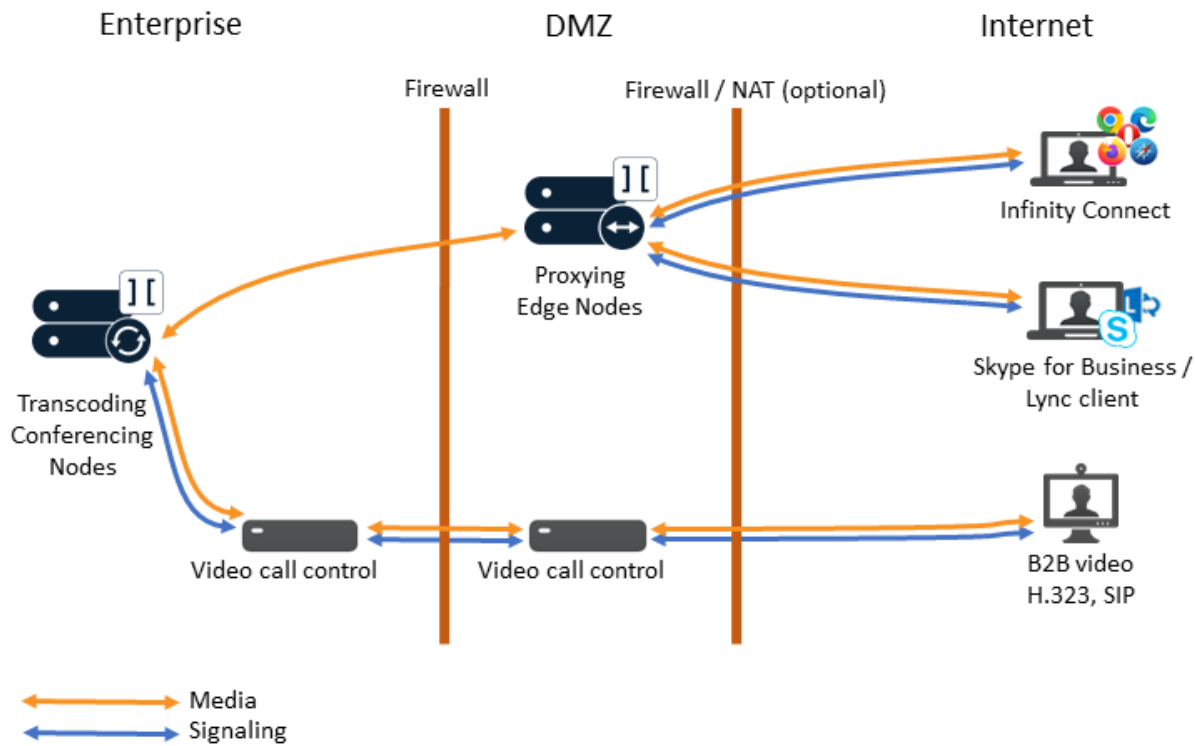
Privately-addressed "on-premises" Transcoding Conferencing Nodes

If you have an on-premises deployment of privately-addressed Pexip Infinity Transcoding Conferencing Nodes, we recommend that you deploy publicly-routable Proxying Edge Nodes to connect external clients and systems to those on-premises transcoding nodes.

The following examples also show how your Pexip Infinity platform may be deployed alongside any existing third-party call control systems and how it can be integrated with on-premises Microsoft Skype for Business / Lync.

Proxying Edge Nodes in combination with third-party video call control

This example deployment scenario uses Proxying Edge Nodes in combination with third-party video call control:



In this type of deployment scenario:

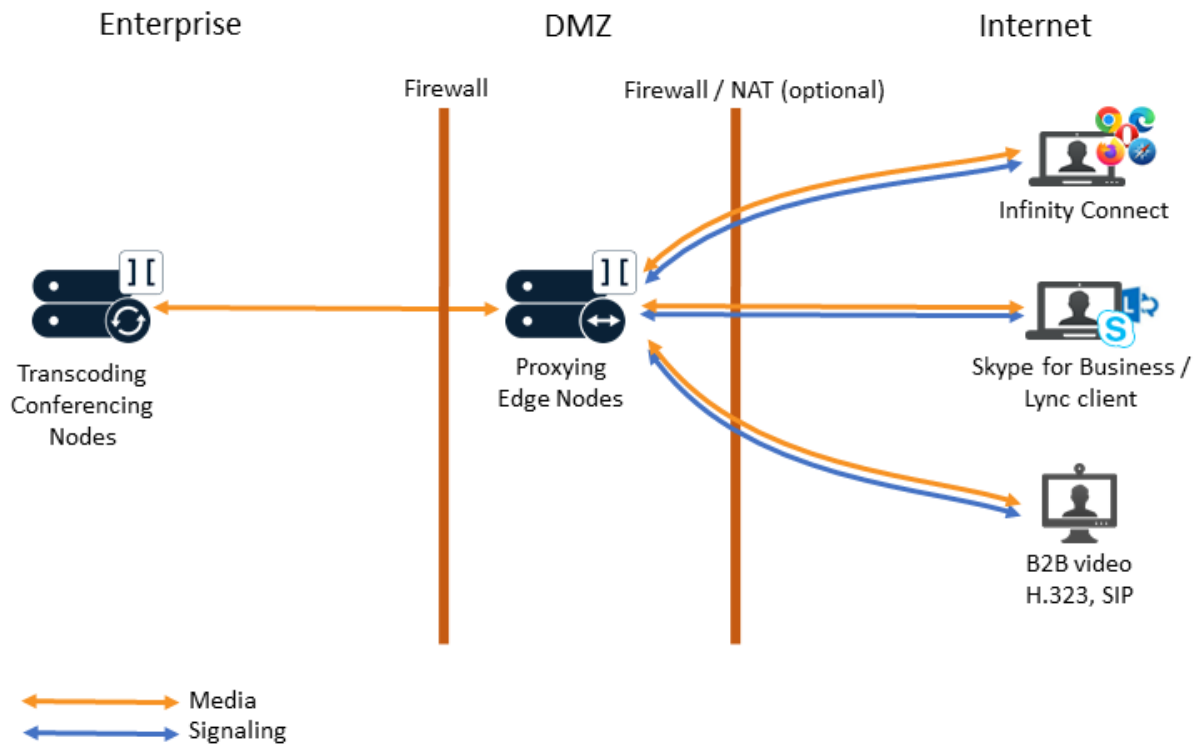
- The Management Node and the Transcoding Conferencing Nodes are deployed with private IP addresses in the local enterprise network.
- External Infinity Connect clients (WebRTC and RTMP) connect to Proxying Edge Nodes. Their signaling is terminated on the proxying node and their media is proxied through to the internal Transcoding Conferencing Nodes. In addition you can optionally deploy a reverse proxy for load balancing and resiliency, or branding.
- Federated Skype for Business / Lync clients connect to Proxying Edge Nodes. Their signaling is terminated on the proxying node and their media is proxied through to the internal Transcoding Conferencing Nodes (an on-premises Skype for Business / Lync server is not needed). As proxying nodes support ICE, a TURN server is not required.
- External SIP and H.323 endpoints and other forms of business-to-business video calls connect to the Transcoding Conferencing Nodes via a firewall traversal / video call control solution such as a Cisco VCS.
- The external firewall can optionally be a NAT device.

For information about integrating Pexip Infinity with other third-party call management systems, see [Integrating Pexip Infinity with other systems](#).

For more information about proxying nodes, see [Deployment guidelines for Proxying Edge Nodes](#).

Routing all external calls via Proxying Edge Nodes

This option extends the previous example by using Proxying Edge Nodes to handle all external calls.



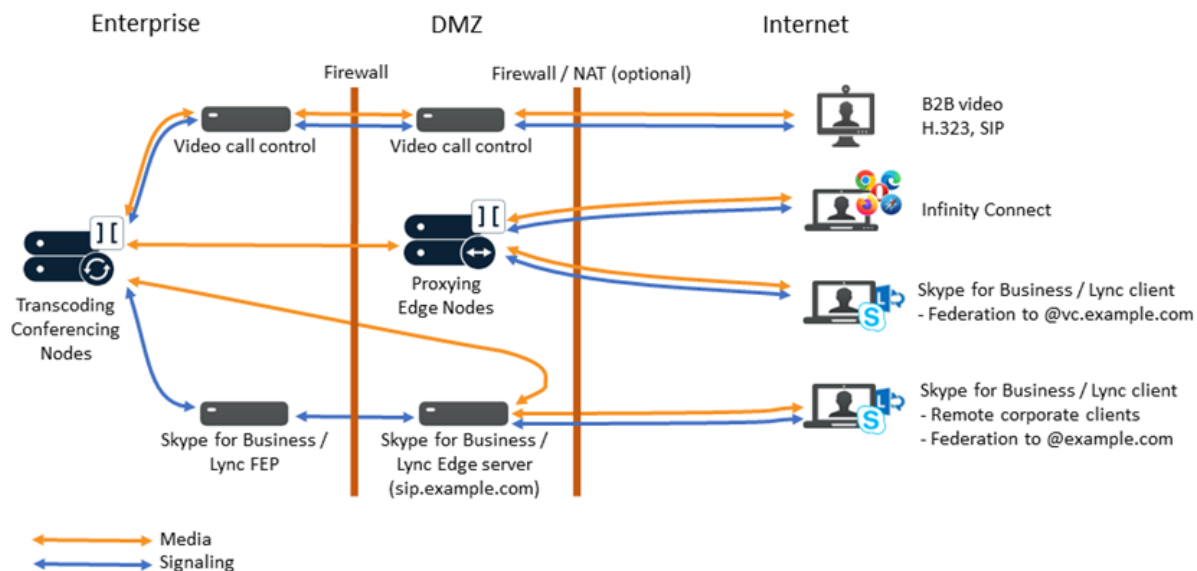
This example deployment scenario is the same as the previous one (with third-party call control), except that:

- SIP and H.323 devices connect directly to Proxying Edge Nodes, which terminate the signaling and proxy their media through to the Transcoding Conferencing Nodes. A third-party call control solution is not required.
- Those SIP and H.323 devices can also register to a Proxying Edge Node if required.

Combining with on-premises Microsoft Skype for Business / Lync

If you have on-premises Microsoft Skype for Business / Lync, you can deploy on-premises Conferencing Nodes alongside your SfB/Lync servers and route external clients as appropriate either via your Proxying Edge Nodes or via your SfB/Lync Edge server.

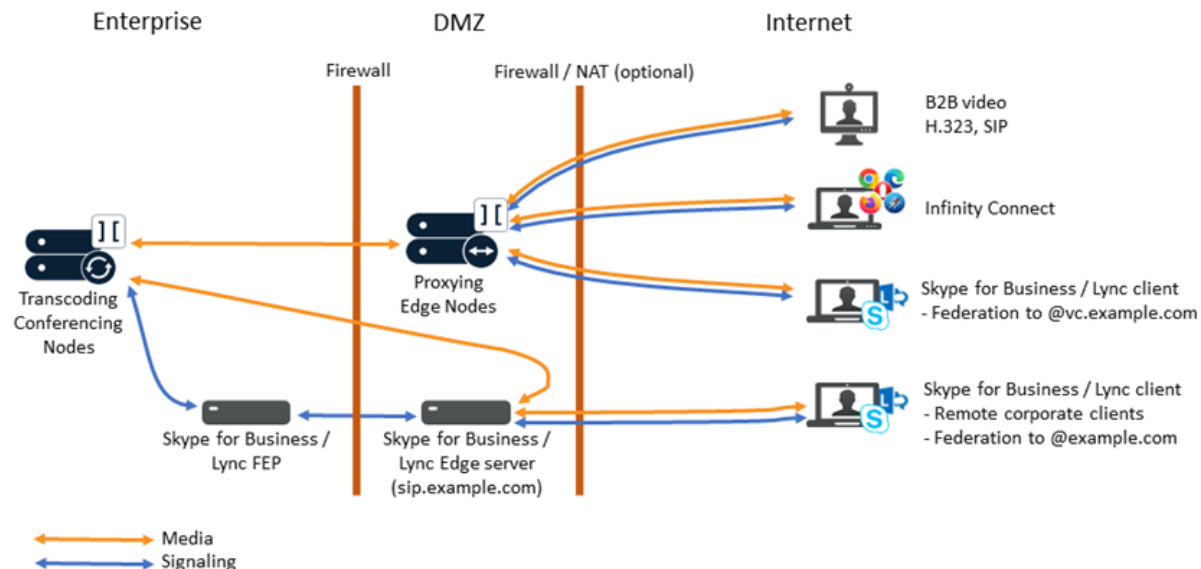
For example your deployment may look like this:



In this deployment scenario:

- The Management Node and the Transcoding Conferencing Nodes are deployed with private IP addresses in the local enterprise network.
- External SIP and H.323 endpoints and other forms of business-to-business video calls connect to the Transcoding Conferencing Nodes via a firewall traversal / video call control solution such as a Cisco VCS.
- External Infinity Connect clients (WebRTC and RTMP) connect to Proxying Edge Nodes. Their signaling is terminated on the proxying node and their media is proxied through to the internal Transcoding Conferencing Nodes. In addition you can optionally deploy a reverse proxy for load balancing and resiliency, or branding.
- Federated SfB/Lync calls to the Pexip Infinity video subdomain (e.g. @vc.example.com) are routed through Proxying Edge Nodes.
- Remote corporate SfB/Lync clients are routed through your Skype for Business / Lync Edge server as normal, but they can also make gateway calls to the Pexip Infinity video subdomain (e.g. @vc.example.com) — in which case media is routed through the SfB/Lync Edge server providing the internal Transcoding Conferencing Node can route to the public facing interface of the SfB/Lync Edge server (otherwise a TURN server is required).
- Federated calls to your SfB/Lync domain (e.g. @example.com) are routed through your Skype for Business / Lync Edge server as normal.
- Calls from external SIP, H.323 and Infinity Connect clients can be gatewayed via Pexip Infinity to SfB/Lync clients or SfB/Lync meetings if required.

Alternatively, your deployment may look like this, which is the same as the one above except that it also routes SIP and H.323 devices through Proxying Edge Nodes, instead of via third-party call control:



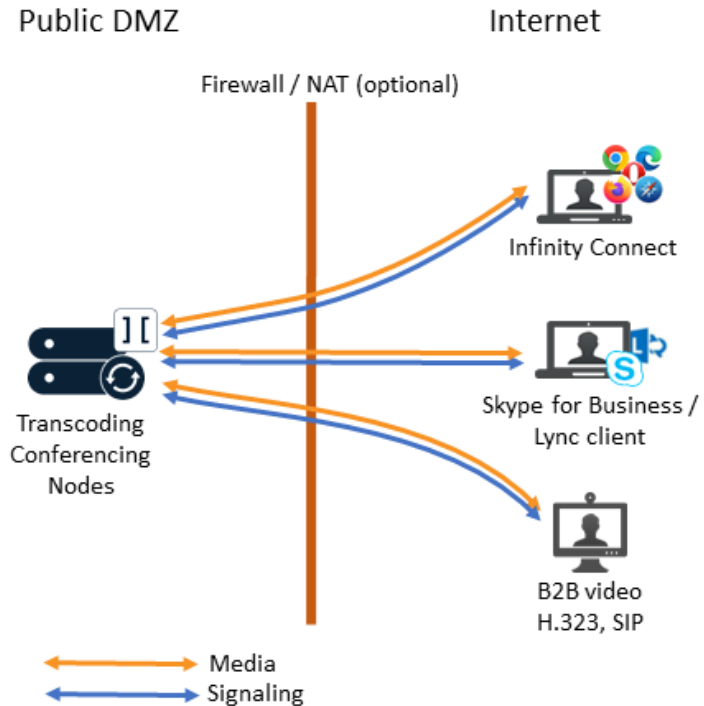
See [Example deployment in an on-prem Skype for Business / Lync environment](#) for more information about deploying Pexip Infinity with on-premises Microsoft Skype for Business / Lync.

Publicly-addressable Transcoding Conferencing Nodes

You can deploy all of your Transcoding Conferencing Nodes in a public DMZ, as shown below.

In this type of deployment:

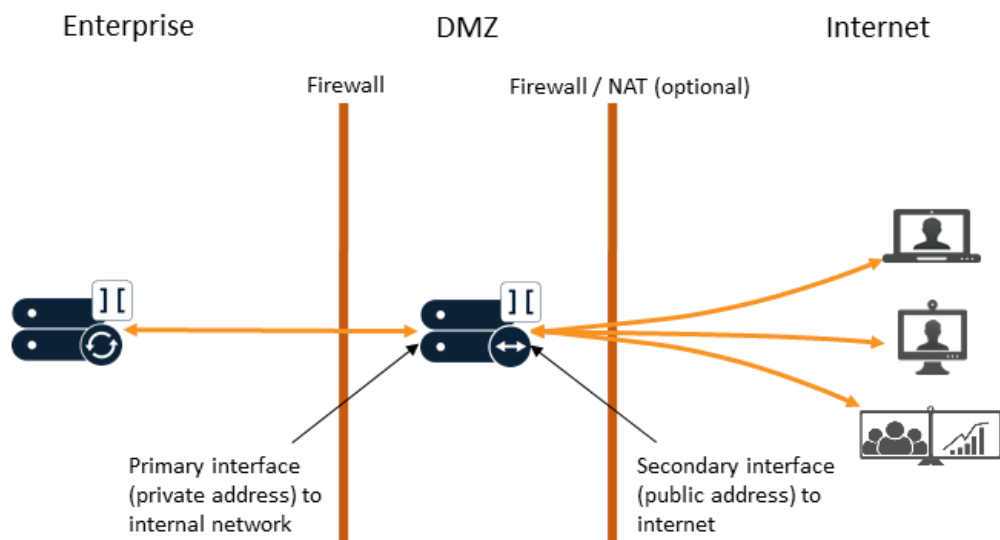
- One or more Transcoding Conferencing Nodes are deployed in a public DMZ. They have publicly-reachable IP addresses — either directly or via static NAT.
- External SIP and H.323 endpoints and other forms of business-to-business video calls can connect directly to the public-facing IP addresses of the Conferencing Nodes.
- Remote federated Skype for Business / Lync clients connect to Pexip Infinity and media can be routed directly through the public-facing IP addresses of the Conferencing Nodes (an on-premises Skype for Business / Lync server is not needed). As Conferencing Nodes support ICE, a TURN server is not required.
- Infinity Connect clients (WebRTC and RTMP) can connect to Pexip Infinity and media can be routed directly to the Conferencing Nodes. In addition you can optionally deploy a reverse proxy for load balancing and resiliency, or branding.
- In large deployments you may want to consider whether to deploy some dedicated Proxying Edge Nodes to manage the signaling and media connections between endpoints situated on the public internet and your Transcoding Conferencing Nodes.
- Note that the Management Node will typically be deployed with a private IP address in the local enterprise network. You must ensure that there is no NAT between the Management Node and the Conferencing Nodes in the DMZ.



Conferencing Nodes with dual network interfaces (NICs)

For additional deployment flexibility, you can configure a secondary network address on a Conferencing Node. Dual NICs are supported on both Transcoding Conferencing Nodes and Proxying Edge Nodes.

You would typically deploy a Conferencing Node with dual network interfaces when it is connected to a dedicated video zone or it is being deployed in a public DMZ where the primary interface would be to an internal, private network segment within the enterprise (where it can connect to the Management Node and other Conferencing Nodes) and the secondary interface would be towards the



video zone or on the publicly-addressable side of the DMZ perimeter network and used for connecting to external endpoints and devices.

When a secondary network address is configured:

- The primary address is always used for inter-node communication to the Management Node and to other Conferencing Nodes.
- SSH connections can be made only to the primary interface.
- The secondary address is always used for signaling and media (to endpoints and other video devices).
- Connections to DNS, SNMP, NTP, syslog and so on, go out from whichever interface is appropriate, based on routing.
- You can have a mixture of any number of single-interfaced and dual-interfaced Conferencing Nodes, providing all nodes can communicate with each other via their primary interfaces.

Note that dual network interfaces are not supported on Conferencing Nodes deployed in public cloud services (Azure, AWS, GCP or Oracle).

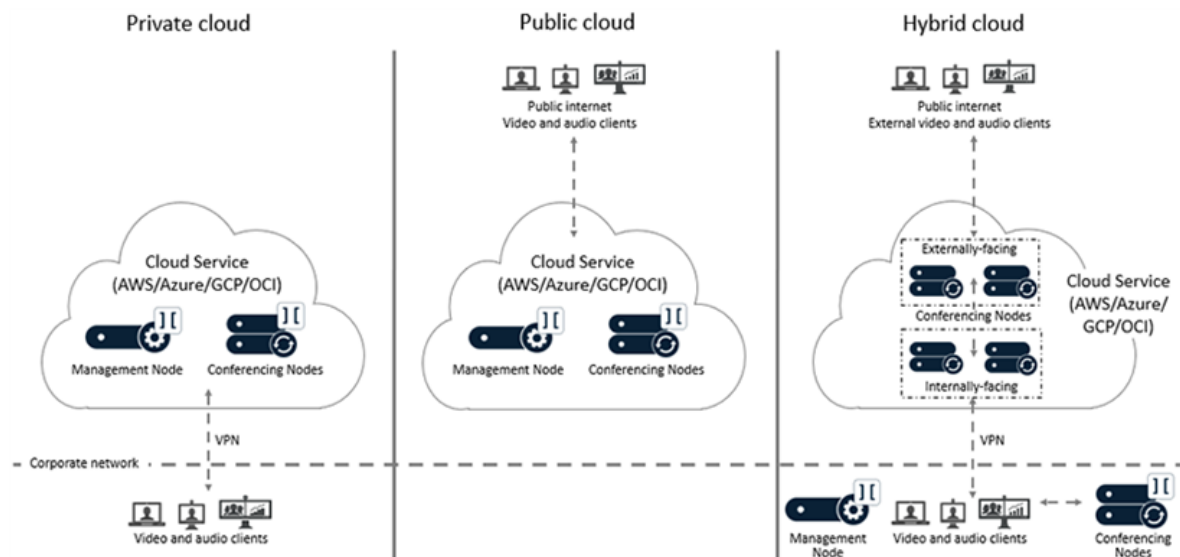
For more information on configuring dual network interfaces, see [Deploying Conferencing Nodes with dual network interfaces \(NICs\)](#) and [Firewall/NAT routing and addressing examples](#).

Deploying as a cloud service via Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP) or Oracle Cloud Infrastructure

The Pexip Infinity platform can be deployed in the Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP) or Oracle Cloud Infrastructure cloud.

There are three main deployment options for your Pexip Infinity platform when using a cloud service:

- **Private cloud:** all nodes are deployed within a private cloud service. Private addressing is used for all nodes and connectivity is achieved by configuring a VPN tunnel between the corporate network and the cloud network. As all nodes are private, this is equivalent to an on-premises deployment which is only available to users internal to the organization.
- **Public cloud:** all nodes are deployed within the cloud network. All nodes have a private address but, in addition, public IP addresses are allocated to each node. The node's private addresses are only used for inter-node communications. Each node's public address is then configured on the relevant node as a static NAT address. Access to the nodes is permitted from the public internet, or a restricted subset of networks, as required. Any systems or endpoints that will send signaling and media traffic to those Pexip Infinity nodes must send that traffic to the public address of those nodes. If you have internal systems or endpoints communicating with those nodes, you must ensure that your local network allows such routing.
- **Hybrid cloud:** the Management Node, and optionally some Conferencing Nodes, are deployed in the corporate network. A VPN tunnel is created between the corporate network and the cloud network. Additional Conferencing Nodes are deployed in the cloud network and are managed from the on-premises Management Node. The cloud-hosted Conferencing Nodes can be either internally-facing, privately-addressed (private cloud) nodes; or externally-facing, publicly-addressed (public cloud) nodes; or a combination of private and public nodes (where the private nodes are in a different Pexip Infinity system location to the public nodes). You may also want to consider dynamic bursting, where the cloud-hosted Conferencing Nodes are only started up and used when you have reached capacity on your on-premises nodes.



All of the Pexip nodes that you deploy in the cloud are completely dedicated to running the Pexip Infinity platform— you maintain full data ownership and control of those nodes.

For specific deployment information about each platform, see:

- [Deploying Pexip Infinity on Amazon Web Services](#)
- [Deploying Pexip Infinity on Microsoft Azure](#)
- [Deploying Pexip Infinity on Google Cloud Platform](#)
- [Deploying Pexip Infinity on Oracle Cloud Infrastructure](#)

Deployment guidelines for Proxying Edge Nodes

A typical deployment scenario is to use Proxying Edge Nodes as a front for many privately-addressed Transcoding Conferencing Nodes. In this scenario, those outward-facing proxying nodes would receive all the signaling and media from endpoints and other external systems, and then forward that media onto other internally-located transcoding nodes to perform the standard Pexip Infinity transcoding, gatewaying and conferencing hosting functions.

The benefit of using only Proxying Edge Nodes to handle all connections from endpoints and other systems is that it can simplify your deployment:

- You only need to set up and maintain DNS records for the proxying nodes.
- As lineside media is terminated at the proxying nodes, you only need to deploy certificates on those proxying nodes.
- You can easily scale up the media processing capacity by adding more transcoding nodes — either in existing locations or by deploying them in new overflow or bursting locations.

In a large Pexip deployment with 5 or more Conferencing Nodes, or where you need to transcode media in multiple locations such as within a DMZ, you should consider deploying Proxying Edge Nodes in addition to your Transcoding Conferencing Nodes. You can easily switch the role of a deployed Conferencing Node from a transcoding node to a proxying node and vice versa.

This topic covers:

- [Configuration summary](#)
- [Configuring a Conferencing Node's proxying or transcoding role](#)
- [Deployment recommendations](#)
- [Deployment scenarios](#)
- [Additional information when deploying Proxying Edge Nodes](#)

Configuration summary

To enable Proxying Edge Nodes on your Pexip Infinity platform:

1. A system location should not contain a mixture of proxying nodes and transcoding nodes. Hence you should decide if you need to create new locations and assign the Conferencing Nodes that you want to use as proxying nodes to those locations:
 - Go to **Platform > Locations** to configure your locations.
 - Go to **Platform > Conferencing Nodes** to change the location assigned to your existing Conferencing Nodes.
 - ❗ If you change the system location of a Conferencing Node, all existing calls will be disconnected and the Conferencing Node will be restarted.
2. Assign a **Role** of *Proxying Edge Node* to those Conferencing Nodes that you want to deploy as proxying nodes:
 - For existing nodes, do this via **Platform > Conferencing Nodes**.
 - When deploying new Conferencing Nodes, you select the **Role** when providing the name and network settings for the new node.
 - ❗ If you change the role of a Conferencing Node, all existing calls will be disconnected and the Conferencing Node will be restarted.
3. Ensure that the locations containing your proxying nodes are **not** configured with a **Transcoding location** of *This location*. Instead, set the **Transcoding location** to a location that contains transcoding nodes.
4. Ensure that your call control systems and DNS are configured to route calls to your proxying nodes only.
5. Ensure that your proxying nodes have appropriate certificates installed (**Platform > TLS Certificates**).
In Microsoft Skype for Business and Lync integrations, your proxying nodes that take federated calls must have a proper certificate. The certificates on any externally facing nodes must be signed by a public CA provider (and thus will be trusted by a Skype for Business / Lync Edge Server). Any internally-facing nodes typically need certificates signed by your private CA.

Full deployment guidelines and example scenarios are described below.

Configuring a Conferencing Node's proxying or transcoding role

Each Conferencing Node is configured with a role — either as a Proxying Edge Node or as a Transcoding Conferencing Node. You specify the Conferencing Node's role when it is first deployed, but you can change its role later if required.

To change a Conferencing Node's role:

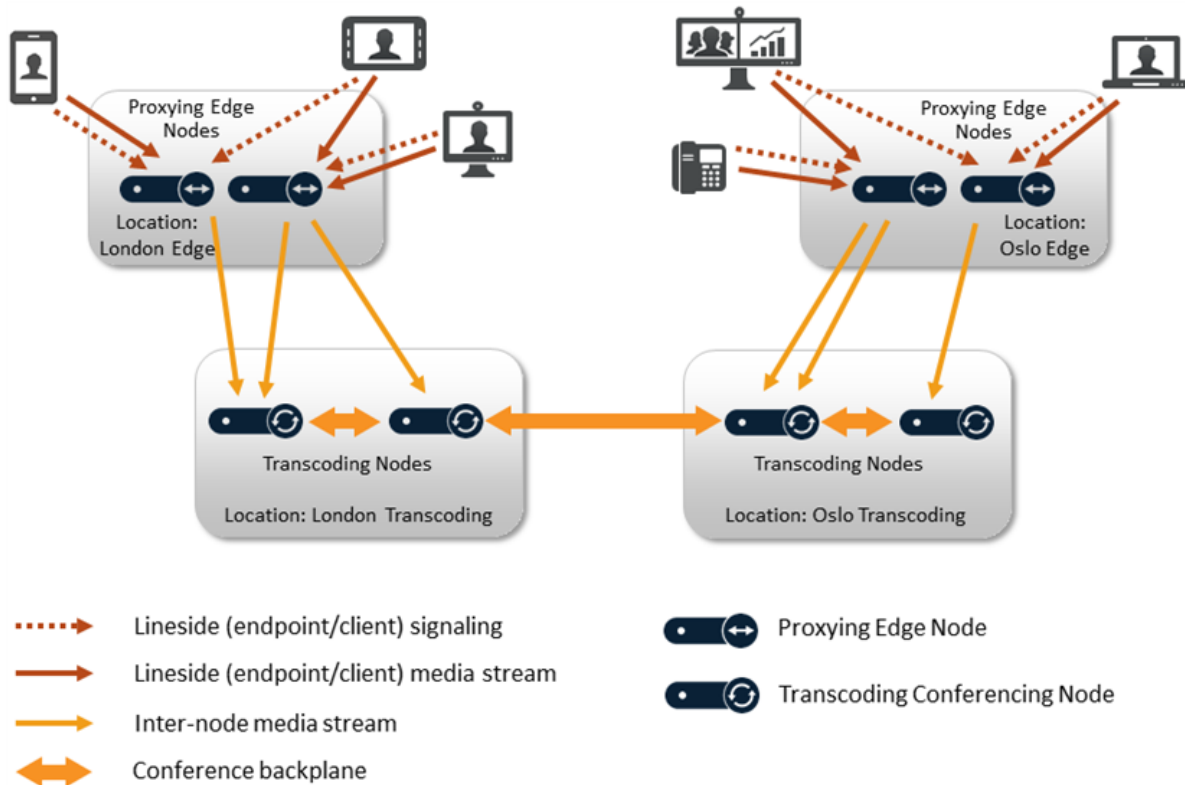
1. Go to **Platform > Conferencing Nodes** and select the name of the Conferencing Node.
2. Select the new **Role** as appropriate and select **Save**.
- ❗ If you change the role of a Conferencing Node, all existing calls will be disconnected and the Conferencing Node will be restarted.
- ❗ If a call is received in a location that contains Proxying Edge Nodes, that location must be configured with a **Transcoding location** that contains your Transcoding Conferencing Nodes.

Deployment recommendations

If you want to deploy Proxying Edge Nodes in your Pexip Infinity platform, we recommend the following guidelines:

- A system location should not contain a mixture of proxying nodes and transcoding nodes. This separation of roles to locations simplifies load-balancing and conference distribution, and makes it easier for you to manage and monitor your externally-facing Proxying Edge Nodes distinctly from your Transcoding Conferencing Nodes. Hence, in the example scenario shown here, the Conferencing Nodes in the two locations "London Edge" and "Oslo Edge" are Proxying Edge Nodes and thus those nodes and locations are not involved in the actual hosting of any conferences. They forward the media onto the Transcoding Conferencing Nodes in the "London Transcoding" and "Oslo Transcoding" locations respectively.
- The location containing your proxying nodes must set its **Transcoding location** to a location that contains your transcoding nodes. You can also optionally specify a primary overflow location and a secondary overflow location for additional transcoding resources i.e. the proxying nodes will proxy to the **Transcoding location** in the first instance, then proxy to the primary and then the secondary overflow locations if the **Transcoding location** runs out of capacity.
- When a location contains Proxying Edge Nodes, those nodes only require IPsec connectivity with:
 - any other proxying nodes in that location
 - all nodes in the transcoding location, and the primary and secondary overflow locations that are associated with that location
 - the Management Node.

This means that the proxying nodes in one location do not need to have a direct network connection to other proxying nodes in other locations.



- Where you have a geographically distributed platform, each physical region should typically have one location containing 1-3 proxying nodes (for resilience and capacity), and one or more additional locations containing transcoding nodes (typically between 2-25 nodes per location depending on capacity requirements).
- Ensure that your call control systems and DNS are configured to route calls to your proxying nodes only, and ensure that your proxying nodes have appropriate certificates installed. (If a call is routed to a transcoding node it can still accept the signaling and handle the call media to and from the endpoint, but we recommend that you defer these functions to your proxying nodes.)
- Lineside media handling is allocated to the proxying node with the most available capacity in the location that receives the signaling (it will **not** allocate the media to proxying nodes in other locations — there is currently no location overflow for proxying nodes). The node that receives the signaling always continues to handle the signaling.
- The proxying node must then forward the media onto a transcoding node. Pexip Infinity's standard media allocation rules are used to decide which transcoding node will receive the proxied media: in the first case it will use a transcoding node in the **Transcoding location** that is associated with the location of the proxying node that is forwarding the media. If there is no capacity to host the conference in the transcoding location then a transcoding node in the primary overflow location, or the secondary overflow location is used (if overflow locations are configured).
- As with all deployment scenarios, there cannot be NAT between any Pexip nodes, but there can be NAT between external devices/internet and proxying nodes. Proxying nodes support all call protocols (Skype for Business / Lync, H.323, SIP, WebRTC and RTMP), and can be deployed with dual network interfaces and static NAT if required.
- You only need to deploy certificates on your Proxying Edge Nodes — only those nodes that handle the signaling connection to an endpoint or other system (such as a Skype for Business / Lync Edge Server) need to be configured with the appropriate certificates to allow that endpoint/system to communicate with Pexip Infinity. If you subsequently deploy more Transcoding Conferencing Nodes to increase conferencing capacity, you do not need to add certificates onto those additional nodes.
- If you use [dynamic bursting](#) to a cloud service, the nodes in your cloud overflow locations should all be Transcoding Conferencing Nodes. You only have to ensure that your proxying nodes can route to your cloud-hosted nodes subnet — as, in this scenario, endpoints will never connect directly to a cloud-hosted node. Note that you cannot increase the your proxying resources via dynamic bursting — you can only have dynamic bursting of transcoding nodes.
- The servers hosting Proxying Edge Nodes do not require as high a specification as those servers hosting Transcoding Conferencing Nodes. This is because proxying nodes are not as processor intensive as transcoding nodes. The minimum functional CPU instruction set for a proxying node is AVX, which was first available in the Sandy Bridge generation. You still need multiple proxying

nodes for resilience and capacity. We recommend allocating 4 vCPU and 4 GB RAM (which must both be dedicated resource) to each Proxying Edge Node, with a maximum of 8 vCPU and 8 GB RAM for large or busy deployments.

Deployment scenarios

Depending on your requirements, there are a range of scenarios where you can make use of a Distributed Edge deployment. These are explained below and range from a basic DMZ Edge scenario to an advanced multi-Edge deployment with dynamic cloud bursting.










- [Basic public DMZ deployment with single NIC Proxying Edge Node](#)
- [Basic public DMZ deployment with dual NIC Proxying Edge Node](#)
- [Basic public DMZ deployment with additional cloud-hosted transcoding resource](#)
- [Multi-Edge deployment with public DMZ and video zone routing](#)
- [Multi-Edge deployment with public DMZ, video zone and PC zone routing](#)
- [Multi-Edge deployment with public DMZ, video zone, PC zone routing and cloud resources](#)
- [Microsoft Skype for Business and Lync integration \(public DMZ / hybrid\)](#)
- [Microsoft Skype for Business and Lync integration \(on-premises\)](#)

Some of these scenarios require routed connections from DMZ nodes — see [Applying static routes to enable routing between externally-facing nodes and local network nodes](#) for more information.

Note that in all Pexip Infinity deployment scenarios:

- The Management Node must be able to reach all Conferencing Nodes (Proxying Edge Nodes and Transcoding Conferencing Nodes) and vice versa.
 - Each Conferencing Node must be able to reach every other Conferencing Node (Proxying Edge Nodes and Transcoding Conferencing Nodes), except:
 - When a location contains Proxying Edge Nodes, those nodes only require IPsec connectivity with:
 - any other proxying nodes in that location
 - all nodes in the transcoding location, and the primary and secondary overflow locations that are associated with that location
 - the Management Node.
- This means that the proxying nodes in one location do not need to have a direct network connection to other proxying nodes in other locations.
- Any internal firewalls must be configured to allow UDP port 500 and traffic using IP protocol 50 (ESP) in both directions between all Pexip nodes.
 - There cannot be a NAT between any Pexip nodes.

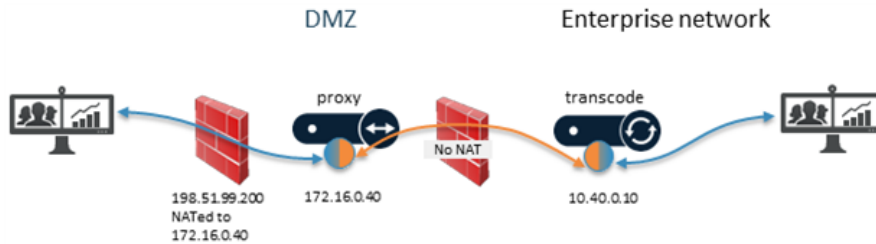
The deployment examples use the following symbols:

Symbol	Description	Symbol	Description
	Lineside (endpoint/client) media path.		Single NIC Proxying Edge Node with or without lineside NAT.
	Pexip to Pexip inter-node media path.		Dual NIC Proxying Edge Node. Blue = line side interface (clients/endpoints). Orange = Pexip internal interface.
	Pexip to Pexip inter-node signaling.		Single NIC Transcoding Conferencing Node with or without lineside NAT.
	Conferencing Node lineside interface (blue dots). These are the only connection points that endpoints/clients see.		Single NIC Transcoding Conferencing Node. Does not (in these scenarios) need to talk to clients as all media is proxied.
	Conferencing Node internal interface (orange dots).		

Basic public DMZ deployment with single NIC Proxying Edge Node

This deployment scenario shows a Proxying Edge Node with a single NIC in a public DMZ. In this example:

- The Proxying Edge Node in the DMZ has a public IP address which is NATted to a private IP address.
- The Proxying Edge Node's private IP address in the DMZ needs a routed connection to the enterprise network (10.40.0.x).
- The Transcoding Conferencing Node in the enterprise network also supports lineside connections to devices in the enterprise network.

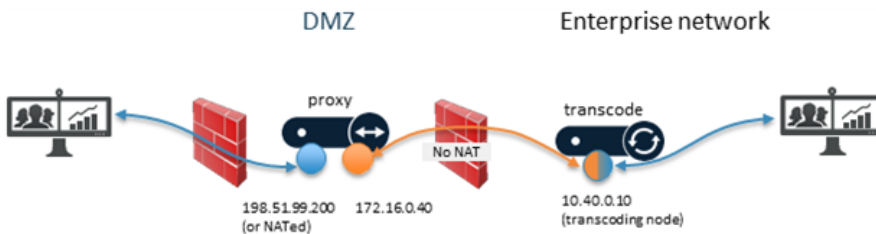


Needs routed connection: 172.16.0.x, 10.40.0.x
Can be separate: 198.51.99.x

Basic public DMZ deployment with dual NIC Proxying Edge Node

This deployment scenario is identical to the previous example, except in this case the Proxying Edge Node has a dual NIC. In this example:

- The secondary NIC on the Proxying Edge Node in the DMZ has a public IP address (which can optionally be NATted).
- The primary NIC on the Proxying Edge Node has a private IP address in the DMZ which needs a routed connection to the enterprise network (10.40.0.x).

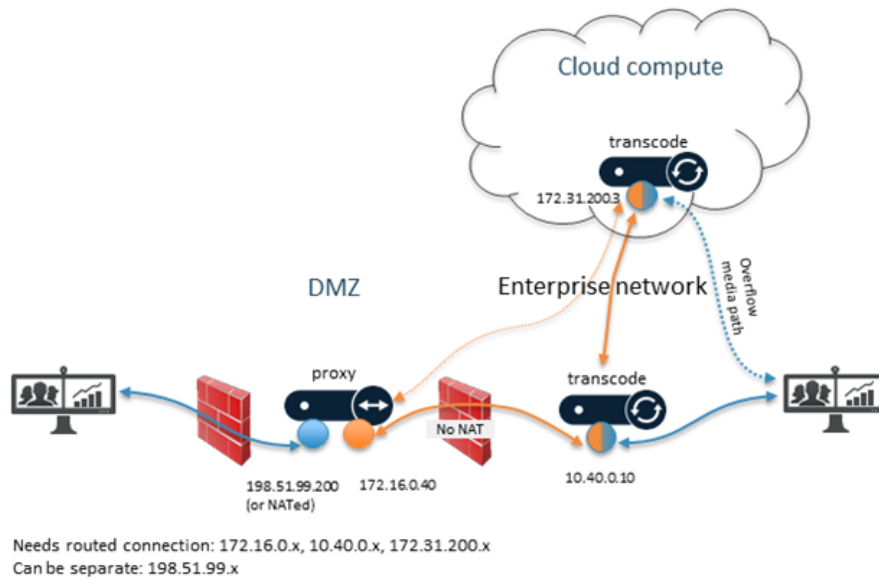


Needs routed connection: 172.16.0.x, 10.40.0.x
Can be separate: 198.51.99.x

Basic public DMZ deployment with additional cloud-hosted transcoding resource

This deployment scenario builds on the previous example by adding additional overflow Transcoding Conferencing Node resources on a cloud-hosted service. In this example:

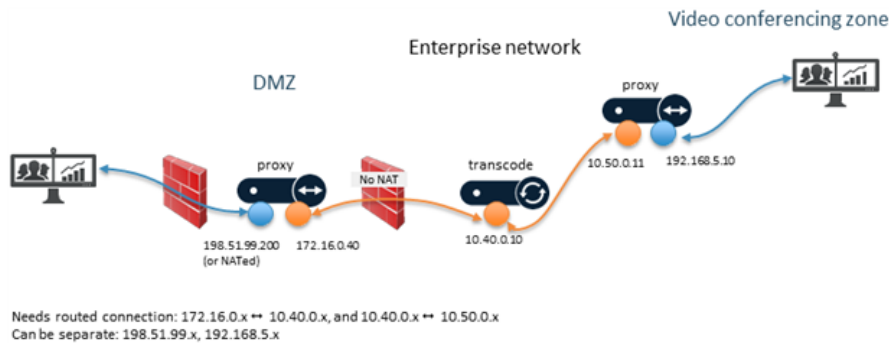
- Enterprise-based devices could get their media assigned to overflow nodes in the cloud service if the on-premises Transcoding Conferencing Nodes are at full capacity.
- The cloud-hosted overflow Transcoding Conferencing Nodes could be "always on" or configured for dynamic bursting.



Multi-Edge deployment with public DMZ and video zone routing

This deployment scenario uses another Proxying Edge Node to handle signaling and media traffic between the enterprise network and the video conferencing zone. In this example we have:

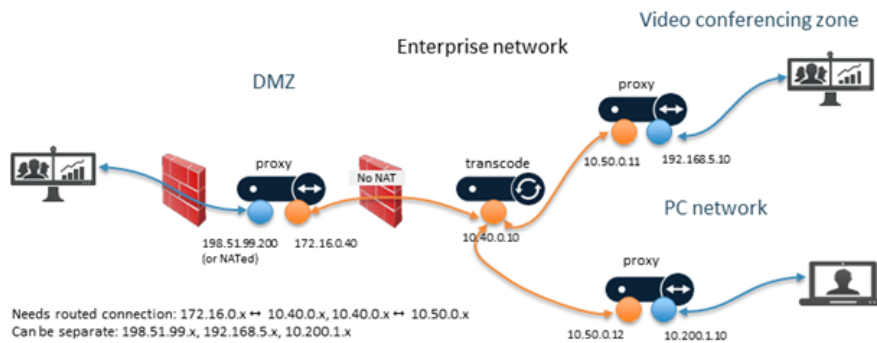
- **Dual NIC Proxying Edge Node in the DMZ:** the secondary lineside NIC has a public IP address (which can optionally be NATted); the primary NIC has a private IP address in the DMZ which needs a routed connection to the enterprise network (10.40.0.x).
- **Dual NIC Proxying Edge Node in the video conferencing network:** the secondary lineside NIC only needs connectivity to the video conferencing zone endpoints (192.168.5.x). The primary internal interface needs a routed connection to the enterprise network.
- A routed connection is not required between proxying nodes in the DMZ and the proxying nodes in the enterprise network.
- No endpoints/clients talk directly to the Transcoding Conferencing Nodes (their media is terminated lineside on Proxying Edge Nodes).



Multi-Edge deployment with public DMZ, video zone and PC zone routing

This deployment scenario adds to the previous example by deploying another Proxying Edge Node to manage connections to a PC network on another internal subnet. Thus, in this example:

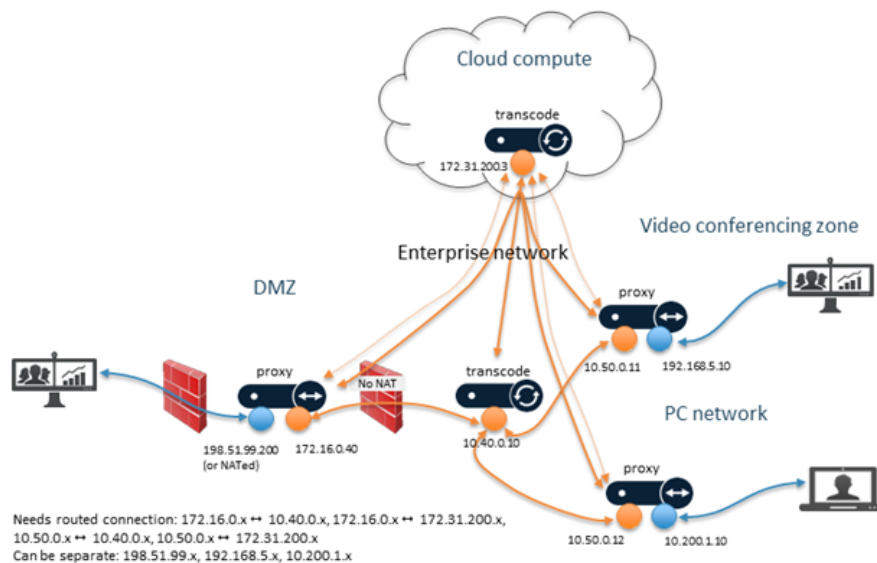
- The private IP addresses in the DMZ (172.16.0.x) and the internal NICs (10.50.0.x) on the proxying nodes handling the video conferencing and PC networks need a routed connection to the enterprise network (10.40.0.x).
- The video conferencing network lineside interface only needs connectivity to video conferencing zone endpoints (192.168.5.x), and the PC network lineside interface only needs connectivity to the PC network (10.200.1.x).
- A routed connection is not required between proxying nodes in the DMZ and the proxying nodes in the enterprise network.



Multi-Edge deployment with public DMZ, video zone, PC zone routing and cloud resources

This deployment scenario brings together all of the elements from the previous examples:

- The private IP addresses in the DMZ (172.16.0.x) and the internal NICs (10.50.0.x) on the proxying nodes handling the video conferencing and PC networks need a routed connection to the enterprise network (10.40.0.x).
- The video conferencing network lineside interface only needs connectivity to video conferencing zone endpoints (192.168.5.x), and the PC network lineside interface only needs connectivity to the PC network (10.200.1.x).
- No endpoints/clients talk directly to the Transcoding Conferencing Nodes in the enterprise network or the cloud-hosted network (their media is terminated lineside on Proxying Edge Nodes). Only the Pexip nodes need to be able to route traffic to the cloud-hosted nodes.
- A routed connection is not required between proxying nodes in the DMZ and the proxying nodes in the enterprise network.



Microsoft Skype for Business and Lync integration (public DMZ / hybrid)

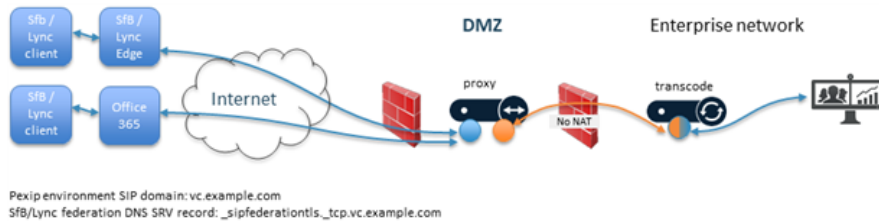
This deployment scenario shows how you can integrate multiple proxying nodes with a public DMZ / hybrid Skype for Business / Lync environment. In this example:

- The Pexip Infinity environment SIP domain is vc.example.com.
- Your Skype for Business / Lync federation DNS SRV record `_sipfederationtls._tcp.vc.example.com` is associated with the hostname `sip.vc.example.com`.
- Round-robin DNS A-records are configured for the `sip.vc.example.com` hostname that point to the IP addresses of your proxying nodes, for example:

Hostname	Host IP address
sip.vc.example.com.	198.51.99.200
sip.vc.example.com.	198.51.99.201

and there are also the "standard" A-records that exist for each proxying node based on their individual hostnames and resolve to the same IP addresses, for example:

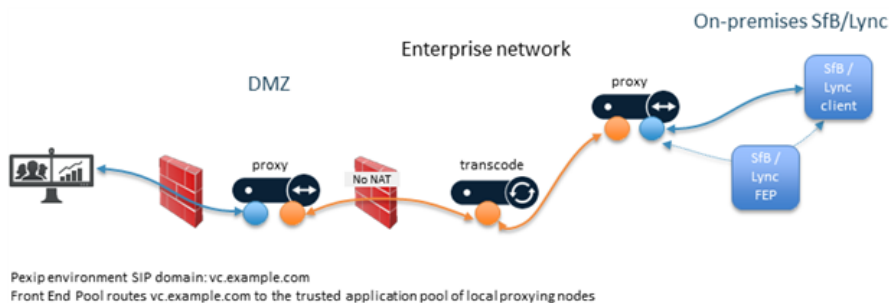
Hostname	Host IP address
proxy01.vc.example.com.	198.51.99.200
proxy02.vc.example.com.	198.51.99.201



Microsoft Skype for Business and Lync integration (on-premises)

This deployment scenario shows how you can integrate internal proxying nodes with an on-premises Skype for Business / Lync environment. In this example:

- The Pexip Infinity environment SIP domain is **vc.example.com**.
- The Skype for Business / Lync environment has a static SIP domain route for the SIP domain **vc.example.com** from the Front End Pool towards a trusted application pool of local proxying nodes.
- Additional proxying nodes are deployed in the DMZ for connectivity with other external devices.



Additional information when deploying Proxying Edge Nodes

- Any Conferencing Node can be a Proxying Edge Node or a Transcoding Conferencing Node, regardless of the hypervisor or cloud platform it is deployed on.
- Even though any media encryption/decryption is performed by a Proxying Edge Node, all subsequent communications with Transcoding Conferencing Nodes is over a secure IPsec connection.
- Proxying Edge Nodes can be deployed anywhere in your network (they do not have to be in a DMZ, for example), but they must be able to forward the media onto at least one Transcoding Conferencing Node.
- Devices can register to Proxying Edge Nodes or to Transcoding Conferencing Nodes.
- A Proxying Edge Node uses less resources per connection than is required by a node that is performing transcoding. A proxying node uses approximately the equivalent of 3 audio-only resources to proxy a video call (of any resolution), and 1 audio-only resource to proxy an audio call.

- Media streams for presentation content to/from an endpoint are forwarded in the same manner as video/audio streams, however they could be handled by different Proxying Edge Nodes (and Transcoding Conferencing Nodes).
- Bandwidth limitations cannot be applied to the forwarding connection between a Proxying Edge Node and a Transcoding Conferencing Node.
- Proxying Edge Nodes do not affect the call licensing requirements for an endpoint connection.

Network routing and addressing options for Conferencing Nodes

This section describes Pexip Infinity's network routing and addressing configuration options that are often required when deploying Conferencing Nodes in a public DMZ, or when routing to a dedicated video zone. It covers the following scenarios:

- [Configuring Pexip Infinity nodes to work behind a static NAT device](#)
- [Applying static routes to enable routing between externally-facing nodes and local network nodes](#)
- [Deploying Conferencing Nodes with dual network interfaces \(NICs\)](#)
- [Remote SIP endpoints behind a remote firewall/NAT](#)

Note that in all Pexip Infinity deployment scenarios:

- The Management Node must be able to reach all Conferencing Nodes (Proxying Edge Nodes and Transcoding Conferencing Nodes) and vice versa.
- Each Conferencing Node must be able to reach every other Conferencing Node (Proxying Edge Nodes and Transcoding Conferencing Nodes), except:
 - When a location contains Proxying Edge Nodes, those nodes only require IPsec connectivity with:
 - any other proxying nodes in that location
 - all nodes in the transcoding location, and the primary and secondary overflow locations that are associated with that location
 - the Management Node.

This means that the proxying nodes in one location do not need to have a direct network connection to other proxying nodes in other locations.

- Any internal firewalls must be configured to allow UDP port 500 and traffic using IP protocol 50 (ESP) in both directions between all Pexip nodes.
- There cannot be a NAT between any Pexip nodes.

In addition, you must ensure that all appropriate firewall ports have been opened as described in [Pexip Infinity port usage and firewall guidance](#).

Further information is available in [Network deployment options](#), [Firewall/NAT routing and addressing examples](#) and [Deployment guidelines for Proxying Edge Nodes](#).

Configuring Pexip Infinity nodes to work behind a static NAT device

To configure your Pexip Infinity deployment to work behind a static NAT device (from the perspective of clients located on the Internet or in a dedicated video zone) you must:

1. Configure the NAT device / firewall with the static, publicly-reachable IP address of each Conferencing Node that you want to be accessible from devices in the internet / video zone, and then map the public address to the node's corresponding internal IP address. Note that it must be a 1:1 NAT.
2. Configure each publicly-reachable Conferencing Node with its **IPv4 static NAT address** (**Platform > Conferencing Nodes**) i.e. the public address of the node that you have configured on the NAT device.

Note that:

- Any Conferencing Nodes that are configured with a static NAT address must not be configured with the same **System location** as nodes that do not have static NAT enabled. This is to ensure that load balancing is not performed across nodes servicing external clients and nodes that can only service private IP addresses.
- Static NAT must be on the secondary interface if the Conferencing Node has dual network interfaces.

- Any internal systems such as Cisco VCSs or endpoints that will send signaling and media traffic to Pexip Infinity nodes that are enabled for static NAT should send that traffic to the public address of those nodes. You must ensure that your local network allows this.
- When integrating with on-premises Microsoft Skype for Business / Lync systems, Conferencing Nodes do not need to use a TURN server for media routing to remote or federated SfB/Lync clients, providing they can reach the public-facing interface of the SfB/Lync Edge server. However, if the Conferencing Nodes are behind a NAT then they do need access to a STUN/TURN server so that each node can discover its NAT address. In Skype for Business / Lync deployments it is essential that a Conferencing Node can discover its NAT address.
- We do not recommend that you allow the Management Node to be accessible from devices in the public internet. However, if you want to do this, you must assign and configure the Management Node with its static NAT address. You should also configure your firewall to only allow access to the Management Node from the specific IP addresses from where you want to allow management tasks to be performed.
- There cannot be a NAT device between any Pexip Infinity nodes.

Deploying Conferencing Nodes with dual network interfaces (NICs)

For additional deployment flexibility, you can configure a secondary network address on a Conferencing Node.

You would typically deploy a Conferencing Node with dual network interfaces when it is connected to a dedicated video zone or it is being deployed in a public DMZ where the primary interface would be to an internal, private network segment within the enterprise (where it can connect to the Management Node and other Conferencing Nodes) and the secondary interface would be towards the video zone or on the publicly-addressable side of the DMZ perimeter network and used for connecting to external endpoints and devices.

As with specifying the IP address of a Conferencing Node with a single interface, the IP address and network mask of the secondary interface must be specified when the Conferencing Node is initially deployed and cannot be subsequently modified.

When a secondary network address is configured:

- The primary address is always used for inter-node communication to the Management Node and to other Conferencing Nodes.
- SSH connections can be made only to the primary interface.
- The secondary address is always used for signaling and media (to endpoints and other video devices).
- Connections to DNS, SNMP, NTP, syslog and so on, go out from whichever interface is appropriate, based on routing.
- You can have a mixture of any number of single-interfaced and dual-interfaced Conferencing Nodes, providing all nodes can communicate with each other via their primary interfaces.
- The Conferencing Node's default gateway can be in either subnet and is automatically associated with the correct interface. For the typical DMZ deployment scenario where the secondary NIC is the public NIC, the default gateway sits on the secondary NIC.
- Static routes can be in either subnet and are automatically associated with the correct interface.
- Static NAT, if configured, must be on the secondary interface.

Applying static routes to enable routing between externally-facing nodes and local network nodes

In some deployment scenarios you may have the Management Node and some Conferencing Nodes deployed in the local enterprise network, and some Conferencing Nodes deployed in a public DMZ or connected to a dedicated video zone.

If the default gateway on those non-local nodes is configured to route traffic out to the internet / video zone, you will need to configure static routes on those Conferencing Nodes to allow them to communicate with Pexip Infinity nodes or other systems in the local, internal network. See [Managing static routes](#) for information about how to configure and assign static routes to Pexip Infinity nodes.

i In situations where the Conferencing Node's default gateway is not towards the Management Node, the static route back to the Pexip Infinity platform must be applied to the Conferencing Node during its initial deployment phase (otherwise it will not be able to communicate with the Management Node and pick up its configuration).

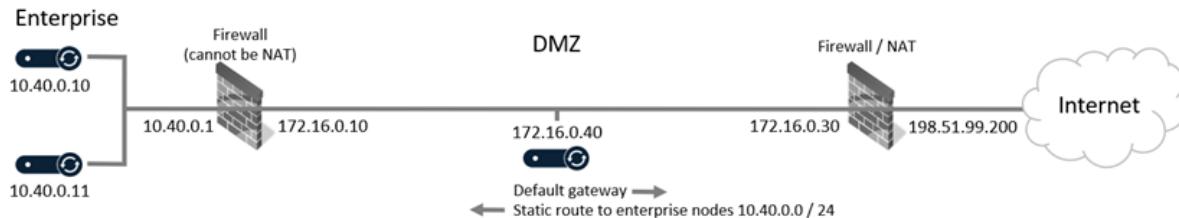
Note that:

- Conferencing Nodes in a DMZ must not be configured with the same **System location** as nodes in a local network. This is to ensure that load balancing is not performed across nodes in the DMZ and nodes in the local network.
- You must configure any internal firewall to allow UDP port 500 and traffic using IP protocol 50 (ESP) in both directions between the Pexip Infinity nodes in the DMZ and the nodes in the local network.

- The firewall between the Pexip Infinity nodes in the DMZ and the nodes in the local network cannot be a NAT device.
- The external firewall between Conferencing Nodes in the DMZ and the internet can be configured with static NAT. In this case you would also need to configure each Conferencing Node in the DMZ with its relevant static NAT address.
- If you deploy the Management Node in the DMZ (although we do not recommend this for security reasons), it must also be assigned with the relevant static route (**Platform > Management Node**).

Example

For example, consider a Pexip Infinity system which is deployed as shown below:



DMZ node has its default gateway at 172.16.0.30 (external firewall)
 DMZ node has static route* to 10.40.0.0 / 24 with gateway IP address of 172.16.0.10
 * Ensure that the static route is applied to the Conferencing Node during its initial deployment phase

Example network with Pexip Infinity nodes in LAN and DMZ

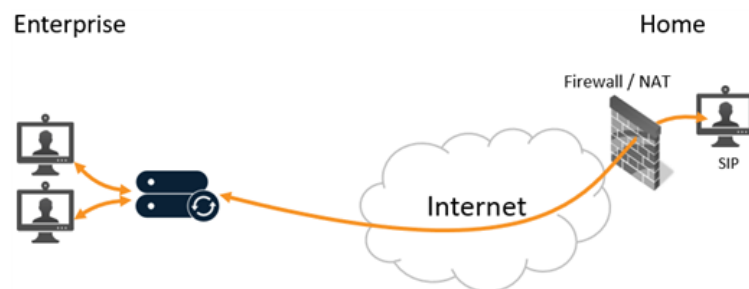
This deployment has:

- Pexip Infinity nodes on the enterprise LAN with addresses in the 10.40.0.0/24 subnet
- an internal firewall (without NAT) with LAN address 10.40.0.1 and DMZ address 172.16.0.10
- an external firewall with DMZ address 172.16.0.30 and public address 198.51.99.200
- a Conferencing Node in the DMZ with a single network interface with address 172.16.0.40 and which is configured with a default gateway address of 172.16.0.30 (the external firewall).

In this situation the Conferencing Node in the DMZ will, by default, send all of its traffic out through its default gateway — the external firewall at 172.16.0.30. To ensure that traffic from the Conferencing Node in the DMZ that is destined for Pexip Infinity nodes on the enterprise LAN can be routed to those nodes, you must deploy the Conferencing Node with a suitable static route applied to it. In this example the **Destination network address** of the static route would be 10.40.0.0 with a **Network prefix** of 24 (to route addresses in the range 10.40.0.0 to 10.40.0.255) and the **Gateway IP address** would be 172.16.0.10 (the DMZ address of the internal firewall).

Remote SIP endpoints behind a remote firewall/NAT

Remote SIP endpoints that are behind remote firewalls/NATs can join Pexip Infinity conferences.



Media latching with remote endpoints behind a remote firewall/NAT

You do not have to apply any explicit configuration to Conferencing Nodes in order to allow remote SIP endpoints behind remote firewalls/NATs to join a conference. Pexip Infinity automatically uses signaling and media port latching to establish routable paths.

(Port latching involves Pexip Infinity waiting until it receives signaling and media traffic from the remote endpoint, and then it uses — or "latches" on to — the source address and port of that traffic as a destination for all traffic bound in the opposite direction. Typically these source addresses/ports will belong to the public interface of the NAT in front of the remote endpoint, and thus anything sent by Pexip Infinity to that address/port should ultimately reach the endpoint.)

Firewall/NAT routing and addressing examples

Here are some example Pexip Infinity routing and addressing scenarios that demonstrate the various [network deployment options](#) that are available when deploying the Pexip Infinity platform:

- [Public DMZ node with single NIC and externally-facing default gateway \(with/without NAT\)](#)
- [Public DMZ node with single NIC and three-legged firewall \(with/without NAT\)](#)
- [Public DMZ node with dual NICs and externally-facing default gateway \(with/without NAT\)](#)
- [Conferencing Node with dual NICs and routing to a video zone \(with/without NAT\)](#)

For more information about how to configure static NAT, dual network interfaces and static routes, see [Network routing and addressing options for Conferencing Nodes](#).

i For specific examples that show how to deploy Proxying Edge Nodes in combination with Transcoding Conferencing Nodes, see [Deployment guidelines for Proxying Edge Nodes](#).

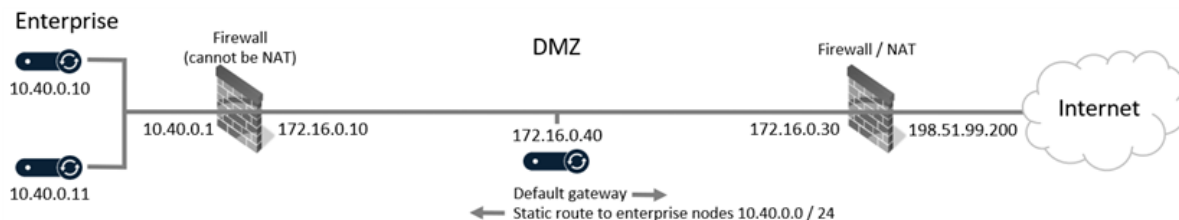
Public DMZ node with single NIC and externally-facing default gateway (with/without NAT)

These types of deployment have a Conferencing Node in the public DMZ with:

- a single externally-facing interface (default gateway is out to the public internet) — this can be with or without NAT
 - a static route back into the enterprise (to communicate with any private Conferencing Nodes and the Management Node)
 - an internal firewall (which must not be NAT) between the enterprise nodes and the public DMZ nodes.
- i** The static route must be applied to the Conferencing Node during its initial deployment phase (otherwise it will not be able to communicate with the Management Node and pick up its configuration).

Externally-facing interface with NAT

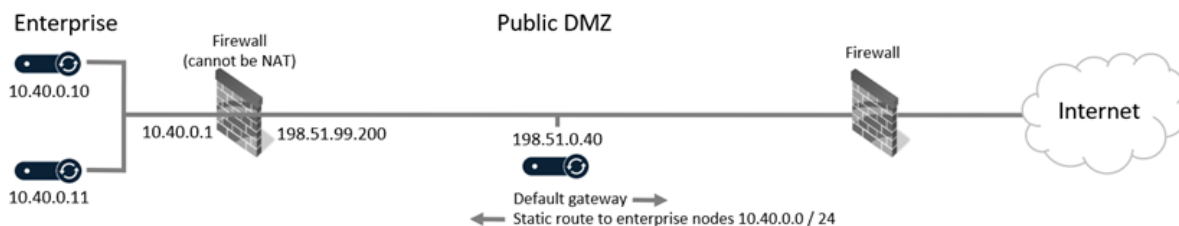
In this example the external firewall performs NAT:



DMZ node has its default gateway at 172.16.0.30 (external firewall)
DMZ node has static route* to 10.40.0.0 / 24 with gateway IP address of 172.16.0.10
* Ensure that the static route is applied to the Conferencing Node during its initial deployment phase

Externally-facing interface without NAT

In this example the external firewall does not perform NAT:



Public DMZ node with its default gateway towards public internet
Public DMZ node has static route* to 10.40.0.0 / 24 with gateway IP address of 198.51.99.200
* Ensure that the static route is applied to the Conferencing Node during its initial deployment phase

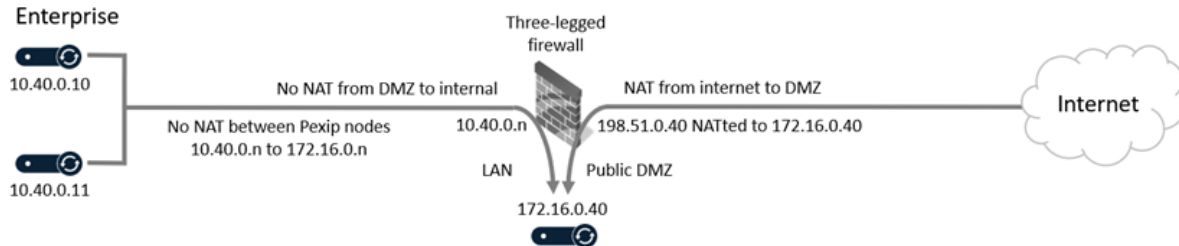
Public DMZ node with single NIC and three-legged firewall (with/without NAT)

These types of deployment have a Conferencing Node in the public DMZ with:

- a single interface — this can be with or without NAT for traffic that is destined for the public internet
- a three-legged firewall that allows bi-directional routing between enterprise nodes in the LAN (any private Conferencing Nodes and the Management Node) — this must always be without NAT.

NAT applied to external traffic

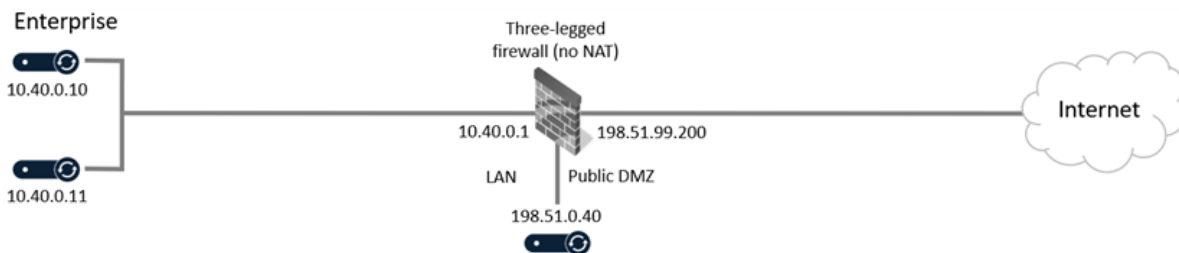
In this example the firewall applies NAT to external traffic (between DMZ and internet):



Firewall allows bi-directional routing between LAN enterprise nodes and public DMZ node. Traffic between DMZ node and internet is NATted.

NAT is not applied to external traffic

In this example the firewall does not apply NAT to external traffic (between DMZ and internet):



Firewall allows bi-directional routing between LAN enterprise nodes and public DMZ node. Traffic between DMZ node and internet is not NATted.

Public DMZ node with dual NICs and externally-facing default gateway (with/without NAT)

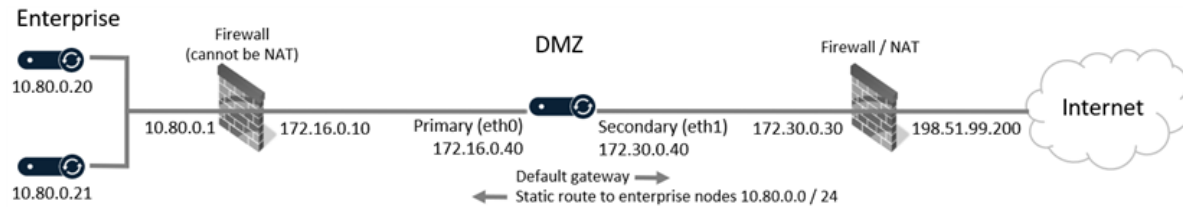
These types of deployment have a Conferencing Node in the public DMZ with:

- dual network interfaces:
 - the primary NIC is internally-facing (eth0 must be used for all inter-node communication)
 - the secondary NIC is externally-facing (eth1 must be used for signaling and media to endpoints and other video devices) — this can be with or without NAT
- the default gateway is out to the public internet (and thus is associated automatically with the secondary NIC)
- a static route back into the enterprise (to communicate with any private Conferencing Nodes and the Management Node)
- an internal firewall (which must not be NAT) between the enterprise nodes and the public DMZ nodes.

i The static route must be applied to the Conferencing Node during its initial deployment phase (otherwise it will not be able to communicate with the Management Node and pick up its configuration).

Externally-facing interface with NAT

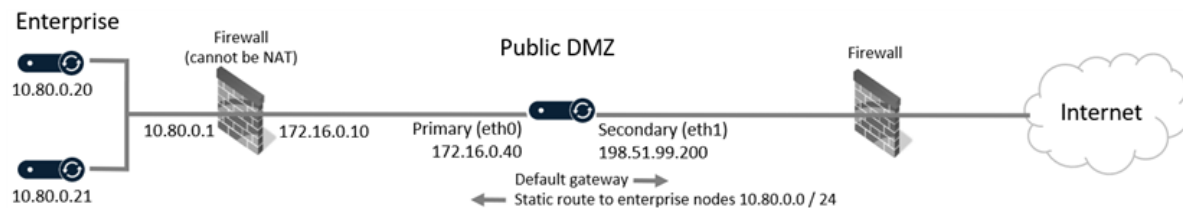
In this example the external firewall performs NAT:



DMZ node has its default gateway at 172.30.0.30 (external firewall) and is associated with secondary interface (eth1)
 DMZ node has static route* to 10.80.0.0 / 24 with gateway IP address of 172.16.0.10
 * Ensure that the static route is applied to the Conferencing Node during its initial deployment phase

Externally-facing interface without NAT

In this example the external firewall does not perform NAT:



Public DMZ node with its default gateway towards public internet
 Public DMZ node has static route* to 10.80.0.0 / 24 with gateway IP address of 172.16.0.10
 * Ensure that the static route is applied to the Conferencing Node during its initial deployment phase

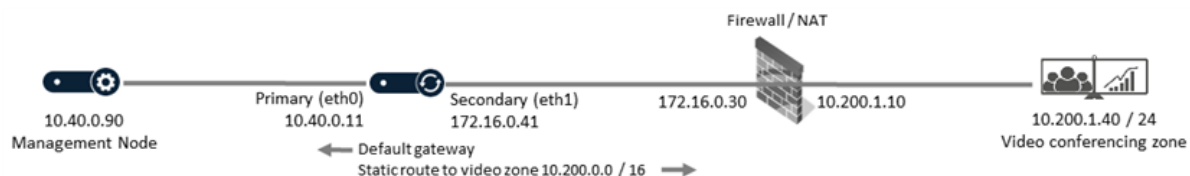
Conferencing Node with dual NICs and routing to a video zone (with/without NAT)

These types of deployment have a Conferencing Node that communicates with a dedicated video conference VLAN. It has:

- dual network interfaces:
 - the primary NIC is internally-facing (eth0 must be used for all inter-node communication)
 - the secondary NIC faces the video zone (eth1 must be used for signaling and media to endpoints and other video devices in the video zone) — this can be with or without NAT
- the default gateway is to the Pexip nodes (and thus is associated automatically with the primary NIC)
- a static route for traffic destined for the video zone.

Video zone routing with NAT

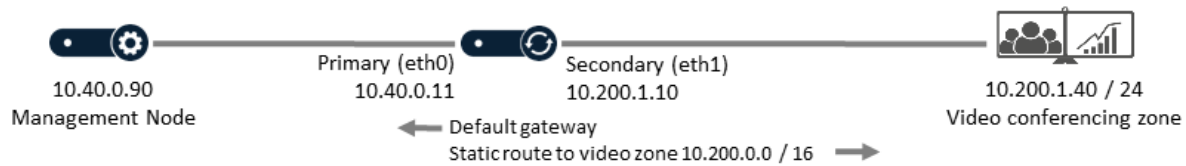
In this example the traffic sent to the video zone has NAT applied:



Conferencing Node default gateway associated with primary interface (eth0) towards Management Node
 Conferencing Node static route to 10.200.0.0 / 16 video zone subnet with gateway IP address of 172.16.0.30

Video zone routing without NAT

In this example the traffic sent to the video zone does not have NAT applied:



Conferencing Node default gateway associated with primary interface (eth0) towards Management Node
Conferencing Node static route to 10.200.0.0 / 16 video zone subnet

Dynamic bursting to a cloud service

Pexip Infinity deployments can burst into the Microsoft Azure, Amazon Web Services (AWS) or Google Cloud Platform (GCP) cloud when principal conferencing capabilities are reaching their capacity limits, thus providing additional temporary Transcoding Conferencing Node resources for media processing.

This provides the ability to dynamically expand conferencing capacity whenever scheduled or unplanned usage requires it. The cloud Conferencing Nodes instances are only started up when required and are automatically stopped again when capacity demand normalizes, ensuring that cloud provider costs are minimized.

You can also use the management API to monitor and manually start up your overflow nodes, if for example, you want to configure a third-party scheduling system to start up all of your additional capacity prior to a large conferencing event starting.

i Cloud bursting provides additional ad hoc capacity in a cloud service to which you must already have a subscription and for which you will be charged based on time used; you must manage the use of this capacity yourself. As an alternative, Pexip also offer [Pexip Smart Scale](#), which provides additional permanent capacity, provided and managed by Pexip.

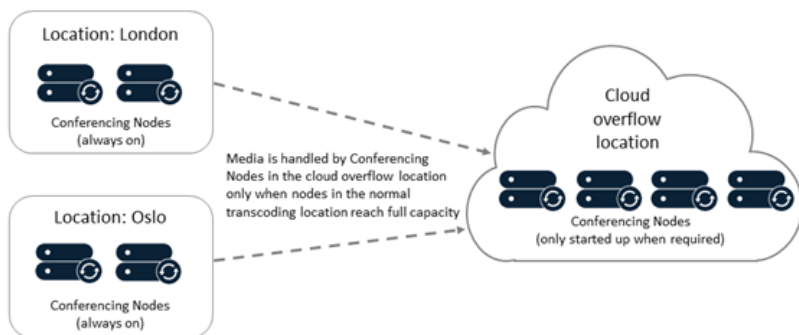
How it works

Dynamic bursting builds upon Pexip Infinity's standard location capacity overflow logic that defines which overflow location's nodes are used when a particular location reaches its media-processing capacity. The dynamic bursting functionality adds to this by automatically starting up those additional cloud nodes when a location is **approaching** full capacity, so that those overflow nodes will be available if required.

After you have deployed your overflow Conferencing Nodes in Azure, AWS or GCP, and enabled cloud bursting and configured your bursting threshold in Pexip Infinity, everything is then controlled automatically by the Management Node:

- Whenever the available capacity in a system location containing your principal (always on) Conferencing Nodes hits or drops below your configured threshold, a Conferencing Node in its overflow location is automatically started up. That new node can then start handling any additional conferencing requirements if the original location reaches full capacity.
- When an overflow Conferencing Node starts to fill up and its location reaches the bursting threshold, a further overflow node is started up, and so on.
- When call levels subside and an overflow Conferencing Node is no longer hosting conferences and is no longer required, the node is automatically shut down again.

This sequence of events is explained in more detail in a [worked example](#).



Configuring your system for dynamic bursting

Dynamic bursting is supported on Conferencing Nodes hosted in Microsoft Azure, Amazon Web Services (AWS) or Google Cloud Platform (GCP). For specific configuration instructions for your chosen platform, see:

- [Dynamic bursting to the AWS cloud](#)
- [Dynamic bursting to the Azure cloud](#)
- [Dynamic bursting to the Google Cloud Platform](#)

Configuring the bursting threshold

When enabling your platform for cloud bursting the most important decision you must make is the level at which to set the bursting threshold:

- The bursting threshold controls when your dynamic overflow nodes in your cloud service are automatically started up so that they can provide additional conferencing capacity. When the number of additional HD calls that can still be hosted in a location reaches or drops below the threshold, it triggers Pexip Infinity into starting up an overflow node in the overflow location.

For example, setting the threshold to 5 means that when there are 5 or fewer HD connections still available in a location, an overflow node will be started up.


- When an overflow location reaches the bursting threshold i.e. the number of additional HD calls that can still be hosted on the Conferencing Nodes in the overflow location reaches the threshold, another overflow node in that location is started up, and so on.

Note that the current number of free HD connections in the original location is ignored when deciding if the overflow location needs to overflow further — however, new calls will automatically use any available media resource that has become available within the original principal location.

- The bursting threshold is a global setting — it applies to every system location in your deployment.
- Note that it takes approximately 5 minutes for a dynamic node instance to start up and become available for conference hosting. If your principal deployment reaches full capacity, and the overflow nodes have not completed initiating, any incoming calls during this period will be rejected with "capacity exceeded" messages. You have to balance the need for having standby capacity started up in time to meet the expected demand, against starting up nodes too early and incurring extra unnecessary costs.

Deployment guidelines

System locations and Conferencing Nodes:

- When configuring your principal "always on" locations, you should normally set the **Primary overflow location** to point at the bursting location containing your overflow nodes, and the **Secondary overflow location** should normally only point at an always-on location.
 -  Nodes in a bursting location are only automatically started up if that location is configured as a **Primary overflow location** of an always-on location that has reached its capacity threshold. This means that if a bursting location is configured as a **Secondary overflow location** of an always-on location, then those nodes can only be used as overflow nodes if they are already up and running (i.e. they have already been triggered into starting up by another location that is using them as its **Primary overflow location**, or you have used some other external process to start them up manually).
- Typically you should assign all of your bursting Conferencing Nodes to a single overflow system location so that all of your main transcoding locations can make use of the same pool of cloud overflow nodes.

However, if you expect a single conference to require more than three overflow nodes, then you could configure a second cloud overflow location (containing a different set of bursting nodes) and then configure half of your principal locations to overflow to the first cloud overflow location, and the other half of your principal locations to overflow to the second cloud overflow location. In this case, both overflow locations will act (start up and shut down nodes as appropriate) independently of each other.
- We recommend that you do not mix your "always on" Conferencing Nodes and your bursting nodes in the same system location.
- Cloud bursting nodes must be deployed as Transcoding Conferencing Nodes (not Proxying Edge Nodes).

Additional guidelines:

- An overflow cloud bursting node is automatically stopped when it becomes idle (no longer hosting any conferences). However, you can configure the **Minimum lifetime** for which the bursting node is kept powered on. By default this is set to 50 minutes, which means that a node is never stopped until it has been running for at least 50 minutes. If your service provider charges by the hour, it is more efficient to leave a node running for 50 minutes — even if it is never used — as that capacity can remain on

immediate standby for no extra cost. If your service provider charges by the minute you may want to reduce the **Minimum lifetime**.

- Do not configure your call control system / DNS to route calls directly to your bursting nodes.
- The Management Node requires access to the cloud platform's APIs to start and stop the bursting node instances. These requests can be routed via a [web proxy](#) if required.
- A location can trigger a bursting node to start up only if there is some existing media load on that location. This is to ensure that incidents such as a temporary network interruption do not inadvertently trigger bursting.
- No specific licenses are required for your bursting nodes, but you must ensure that your overall system has sufficient licenses to meet peak conference capacity demand.
- Your cloud bursting nodes must all be running on the same cloud platform (Azure, AWS or GCP).
- All log messages that are explicitly related to cloud bursting, such as starting up or shutting down overflow nodes, are tagged with a log module name of administrator.apps.cloudbursting. All other log messages related to those overflow nodes or the conferences they are hosting are reported in the same manner as per standard behavior.

Detailed example of the overflow process

This sequence of actions listed below shows how the process of starting and stopping overflow nodes is managed. It assumes an example scenario where:

- the system is configured with a bursting threshold of 5
- there is a single principal location (London) containing 2 "always on" Conferencing Nodes with a total location capacity of 40 connections
- the principal location is configured to overflow to the cloud overflow location "Bursting Europe"
- the "Bursting Europe" location contains 2 overflow bursting nodes, each with a capacity of 20 connections.

Principal location: London			Cloud overflow location: Bursting Europe		
Action	Call capacity remaining	Dynamic bursting activity	Overflow node A status	Overflow node B status	Call capacity remaining
1. No conferences in progress	40	<i>none</i>	Not running	Not running	
2. A conference starts and has 33 participants	6*	<i>none</i>			
3. 1 more participant joins	5	Principal location threshold reached	Node A starting		
4.	5	Overflow capacity becomes available	Running		20
5. 5 more participants join	0	<i>none</i>	↓		20
6. 1 more participant joins	0	Call media handled by overflow node A	↓		18*
7. 12 more participants join	0	All new participants handled by overflow node A	↓		6
8. 1 more participant joins	0	Overflow location threshold reached	↓	Node B starting	5
9.	0	Additional overflow capacity becomes available	↓	Running	25
10. 7 more participants join	0	All new participants handled by overflow nodes A and B	↓	↓	17*

Principal location: London			Cloud overflow location: Bursting Europe		
Action	Call capacity remaining	Dynamic bursting activity	Overflow node A status	Overflow node B status	Call capacity remaining
11. Conference ends	40	Overflow nodes remain available for further bursting if required	↓	↓	40
12. A new conference starts with 25 participants	14*	<i>none</i>	↓	↓	40
13.		Overflow node A is unused and has been running for 50 minutes	Shutting down	↓	20
14.		Overflow node B is unused and has been running for 50 minutes	Not running	Shutting down	0
15. Conference ends	40		Not running	Not running	

* A Conferencing Node reserves 1 HD connection for the backplane to other nodes in the same conference.

Note that if other conferences were running on any of the nodes in these locations, they would consume call capacity and bursting would be triggered in exactly the same way when the remaining capacity within the location reached the threshold.

Viewing cloud bursting status

Current bursting status

Go to **Status > Cloud Bursting** to see an overview of the media load of your principal locations (that contain your "always-on" Conferencing Nodes), and whether your overflow nodes and locations are in use.

- Any issues relating to your cloud bursting deployment will also be shown on this page.
- The list of principal locations only includes those locations that are configured with a **Primary overflow location** that contains bursting nodes.
- An **approaching threshold** message is displayed in the **Available HD connections** column for the principal locations when the number of available HD connections is less than or equal to the bursting threshold plus two.
This message changes to **bursting threshold reached** when the number of available HD connections is less than or equal to the bursting threshold (and therefore overflow nodes are started up).
- You can manually start any overflow nodes by selecting **Start** for the required node (the **Start** option is in the final column of the **Cloud overflow nodes** table).
- The status page dynamically updates every 15 seconds.

Bursting history

Go to **Status > Conferencing Node History** to see all of the events (stop, start or running) that have been applied to overflow Conferencing Nodes and, where appropriate, the reason why the event was applied (for example if a node was shut down as there was no longer a need for the extra capacity).

Handling of media and signaling

Media and signaling for each call to a Pexip Infinity service (VMR, gateway call etc.) may take different paths, depending on the location of the caller, the available capacity of Conferencing Nodes, whether any media overflow locations have been configured, and the role of the Conferencing Node that receives the call signaling.

A Conferencing Node can have either a transcoding or a proxying role:

- Transcoding Conferencing Nodes are required in all deployments; they manage all of the media processing required to host a conference, and can also handle direct connections to/from endpoints if required.
- Proxying Edge Nodes are optional; they handle call signaling and the media connection with the endpoint, but forward the media on to a Transcoding Conferencing Node for processing. For full information, see [Deployment guidelines for Proxying Edge Nodes](#).

Behavior summary

Media and signaling handling for a conference follows these rules:

- A single conference instance can span any number of locations and any number of Conferencing Nodes, but with a limit of **3 nodes per location** that are processing media for that conference.
 - The Conferencing Node that receives the call signaling always continues to handle the signaling, regardless of where or how the media is routed (which may be through a different node). There is no concept of signaling overflow.
 - When a call is received on a Transcoding Conferencing Node, Pexip Infinity selects a transcoding node to process the conference media and perform the lineside media handling. The selection process is based on the **Transcoding location**, and the optional **Primary overflow location** and **Secondary overflow location** configured against the location of the node that is handling the call signaling. In the first case it tries to use a transcoding node in the **Transcoding location**: it will initially select a node that is already transcoding media for that conference (if that node has sufficient capacity to take the new call), otherwise it selects the transcoding node that currently has the most available capacity. If there is not a suitable node in the **Transcoding location** with sufficient capacity to take the media, or it has reached the limit of 3 nodes in that location, it uses a transcoding node in the **Primary overflow location**, or a node in the **Secondary overflow location**.
 - When a call is received on a Proxying Edge Node, lineside media handling is allocated to the proxying node with the most available capacity in the location receiving the signaling. That proxying node must then forward the media onto a transcoding node using the same allocation/overflow rules (based on the signaling location's configured transcoding and media overflow locations) as when a call is received on a transcoding node, as described above. A system location should not contain a mixture of proxying nodes and transcoding nodes.
 - The transcoding nodes send the call media for the conference to each other over a backplane, with each node sending the media on behalf of all the endpoints connected (or proxied) to it. If two or more transcoding nodes in the same location are processing conference media, a single node per location acts as an intermediary and sends the conference media over a backplane to another intermediary node in the other locations that are also handling that conference.
- i** You should ensure that any locations specified as a **Transcoding location** or as a **Primary** or a **Secondary overflow location** contain only Transcoding Conferencing Nodes.

Note that if you change the media-handling locations for a proxying location (i.e. its transcoding, primary or secondary overflow locations), any proxied calls from that location that are **currently** being handled by the previously configured media locations will be dropped.

More details about all of these rules and some example scenarios are explained in the sections below:

- [Locally and globally distributed conferences](#)
- [How nodes for signaling and media proxying \(if applicable\) are determined](#)
- [How Pexip Infinity decides which Transcoding Conferencing Node will process the media](#)
- [Media overflow locations](#)
- [Infinity Gateway calls](#)
- [Nominating the outgoing location for outbound calls](#)
- [Examples](#)

Locally and globally distributed conferences

Conferences that span more than one Transcoding Conferencing Node can be locally or globally distributed. **Locally distributed conferences** exist across multiple transcoding nodes in the same system location and send conference media between nodes via a local backplane.

Globally distributed conferences exist across several Transcoding Conferencing Nodes, where each node is in a different system location. As system locations are typically used to represent different physical locations, this allows participants in different regions to access the conference from their local Conferencing Node. The nodes send the call media for the conference to each other over a single geo backplane, with each node sending the media on behalf of all the endpoints connected (or proxied) to it, thus minimizing WAN bandwidth usage between locations.

A conference can be **locally and globally distributed** at the same time, if two or more Transcoding Conferencing Nodes in one location and at least one other transcoding node in a different location are involved. In such cases, one transcoding node in each location acts as the intermediary for any other transcoding nodes in the same location that are handling the media for that conference. Call media for each location is sent between the intermediaries only, thus minimizing WAN bandwidth usage between locations.

A single conference instance can span any number of locations and any number of transcoding nodes, but with a maximum limit of **3 nodes per location** that are performing media transcoding for that conference.

How nodes for signaling and media proxying (if applicable) are determined

In all cases, when a call is placed to a Pexip Infinity service, the call (signaling) is received by whichever Conferencing Node was selected by your call control system or routed via DNS.

- Signaling always remains routed to the node that received the call. This could be a Proxying Edge Node or a Transcoding Conferencing Node.
- If the signaling is received on a Proxying Edge Node, media proxying is allocated to the proxying node with the most available capacity in the location that received the signaling. The selected proxying node will always handle the media connection with the endpoint, acting as a proxy between the endpoint and a Transcoding Conferencing Node (which should be in a different location).
- If the signaling is received on a Transcoding Conferencing Node, then whichever transcoding node is selected to process the media (which may be a different node to the signaling node) will also directly handle the media connection with the endpoint.

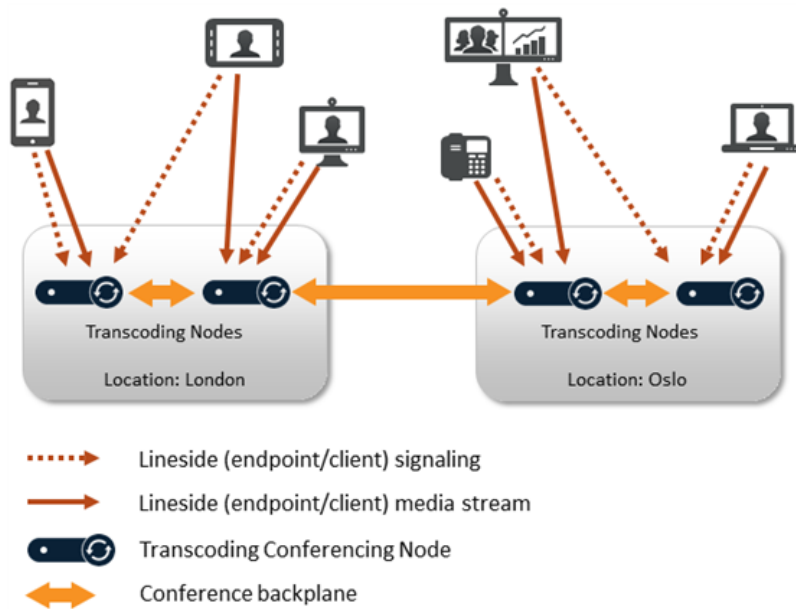
How Pexip Infinity decides which Transcoding Conferencing Node will process the media

Pexip Infinity intelligently load-balances the call media across all transcoding nodes that are grouped within a system location. It uses the same load-balancing rules to decide which transcoding node will process the media and host the conference, regardless of whether the media is being proxied or not, and whether the signaling is being handled by a proxying node or by a transcoding node:

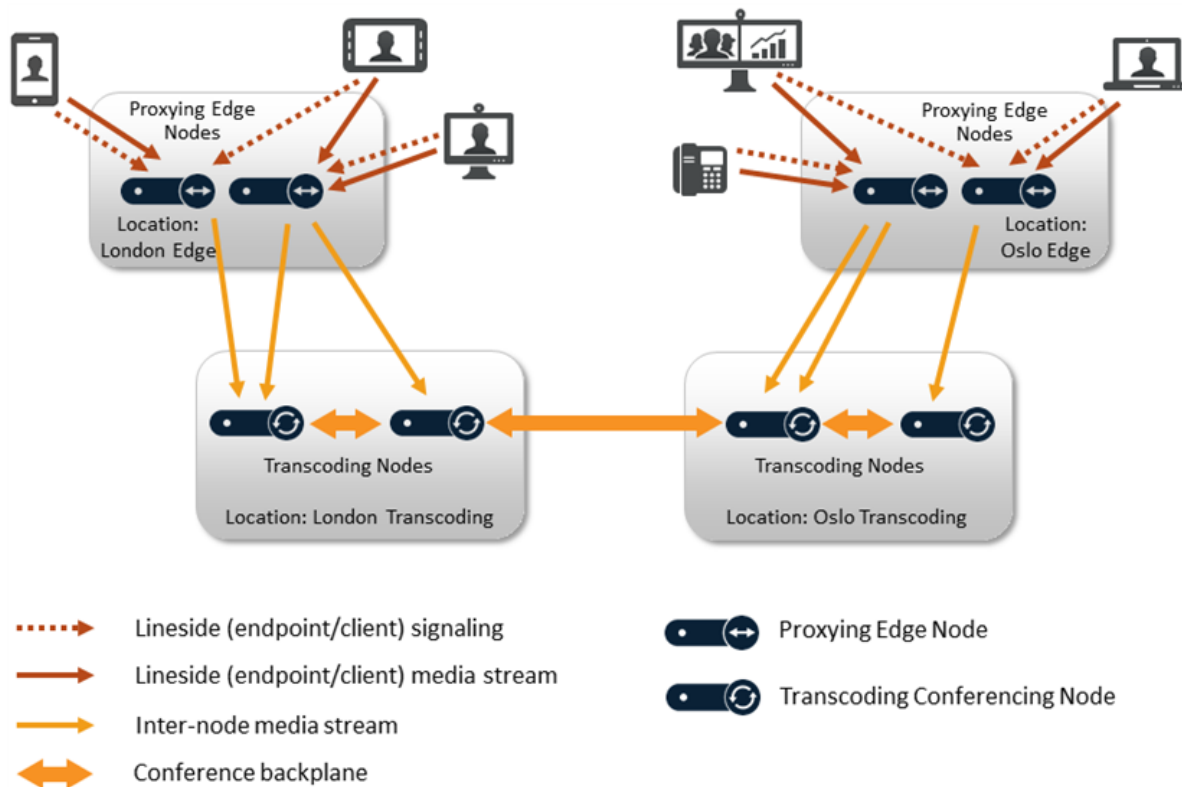
- The selection process is based on the **Transcoding location**, and the optional **Primary overflow location** and **Secondary overflow location** configured against the location of the node that is handling the call signaling.
- Initially, Pexip Infinity tries to use a Transcoding Conferencing Node that is in the **Transcoding location**:
 - If there is a transcoding node that is already handling media for the conference, and it has spare capacity, then that node will also process the media on the new call.
 - If there are no transcoding nodes that are currently processing media for the conference, or those that are currently processing media for the conference are at full capacity, then the transcoding node in that location that currently has the most available capacity is selected (up to a maximum of three transcoding nodes per location per conference instance).
- If there is **no transcoding resource available in the Transcoding location**, then a transcoding node in a **media overflow location** (if configured) is used. The reasons why there may be no transcoding resource available in the **Transcoding location** are:
 - all of the transcoding nodes in that location are fully-loaded in hosting other conferences
 - there are three fully-loaded transcoding nodes already managing that conference in that location
 - all of the nodes in the **Transcoding location** are proxying nodes (for example, if a location containing proxying nodes has its **Transcoding location** set to *This location*).

The selection of the transcoding node in the media overflow location follows the same balancing rules: if a node is currently processing the conference media and has capacity to take the new call then that node is selected, otherwise the node in the overflow location that currently has the most available capacity is selected.

- The call is rejected if there is no transcoding capacity available in the transcoding location or either of the overflow locations associated with the signaling location. Note that endpoints connected (for signaling purposes) to nodes at other locations may still be able to join the conference — this is because each location has its own configurable set of transcoding and overflow locations that it can use for its local endpoints.



Example flows when only using Transcoding Conferencing Nodes and transcoding is performed in the same location as the signaling location



Example flows when using Proxying Edge Nodes and transcoding is performed in a separate location containing Transcoding Conferencing Nodes

Media overflow locations

Media overflow locations are used to tell Pexip Infinity where to process the call media when there is no transcoding resource currently available in the Transcoding location associated with the location where the call has been received.

Locations are not explicitly configured as "media overflow" locations — their use depends upon how you have configured your system. For example, you could have one dedicated location that never initially handles any calls, but is used as a common overflow location

for all of your other locations. Alternatively, you could configure your locations to overflow to each other — which may be useful if you have locations around the world, so that a location that is experiencing heavy demand during its working day can make use of idle resources in a location in a different timezone.

The nodes in your overflow location can be "always-on" nodes or you can use dynamic node resources in a cloud service (referred to as "[bursting](#)").

To enable overflow, you must configure a location with a **Primary overflow location** and a **Secondary overflow location** (in addition to its standard **Transcoding location**). When overflow resources are required, Pexip Infinity will use any available transcoding resource in the primary overflow location first (using the same rules and limitations as described above), and if there is no suitable resource in the primary overflow location it will use resource in the secondary overflow location.

If you need to nominate more than two overflow locations, you can use local or external policy to specify a list of multiple overflow locations in its response to a media location request.

i When determining where to handle the media for any given call, the transcoding location, primary overflow location and secondary overflow location options are all based on what is configured directly against the location containing the Conferencing Node that is handling the signaling for that call — i.e. if an overflow location is configured with its own overflow locations, then those indirectly-configured overflow locations are ignored. For example, if a call is received in location A, and location A is configured to transcode in location B but location B is full, it next uses the overflow location as configured in location A (what is configured in location B is irrelevant to this call). See the [examples](#) below for more details and conferencing scenarios.

Note that:

- If you are not using proxying nodes, you must ensure that endpoints can route their media to the transcoding nodes in the overflow locations.
- When each new location is brought into the conference, Pexip Infinity nominates a single transcoding node in each location as the intermediary node for that conference instance, and a geo backplane is established between those intermediary nodes.

Infinity Gateway calls

The same principles described above also apply to calls placed via the Infinity Gateway:

- The call signaling is always handled by the Conferencing Nodes that receive and place the call (note that you can configure Call Routing Rules to call out from a Conferencing Node in a different location to that which received the call).
- The call media is handled by a transcoding node that has the most available capacity in the **Transcoding location** associated with the signaling node's location, or its overflow location. However, see below for additional information about the outgoing location.

Nominating the outgoing location for outbound calls

When specifying the outgoing location for a Call Routing Rule, automatically dialed participant, or when manually dialing out from a conference:

- The call (signaling) is placed from a Conferencing Node in the nominated outgoing location, unless the device being called is registered to Pexip Infinity, in which case the call is always placed from the Conferencing Node where the registration is held.
- If the node placing the call is a Proxying Edge Node, then the proxying node in that signaling node's location with the most available capacity will handle the lineside media connection with the called device and it will proxy the media to a transcoding node in the **Transcoding location** associated with that location (or to an overflow location if there is no transcoding capacity in the **Transcoding location**).
- If the node placing the call is a Transcoding Conferencing Node, then the transcoding node in the **Transcoding location** associated with that signaling node's location (or in an overflow location if there is no transcoding capacity in the **Transcoding location**) with the most available capacity will handle the lineside media connection and the media transcoding.
- Note that if the nominated outgoing location contains a mixture of proxying nodes and transcoding nodes (although we do not recommend this) then Pexip Infinity will choose any node from the location and it will behave as described above according to the selected node's role.

Examples

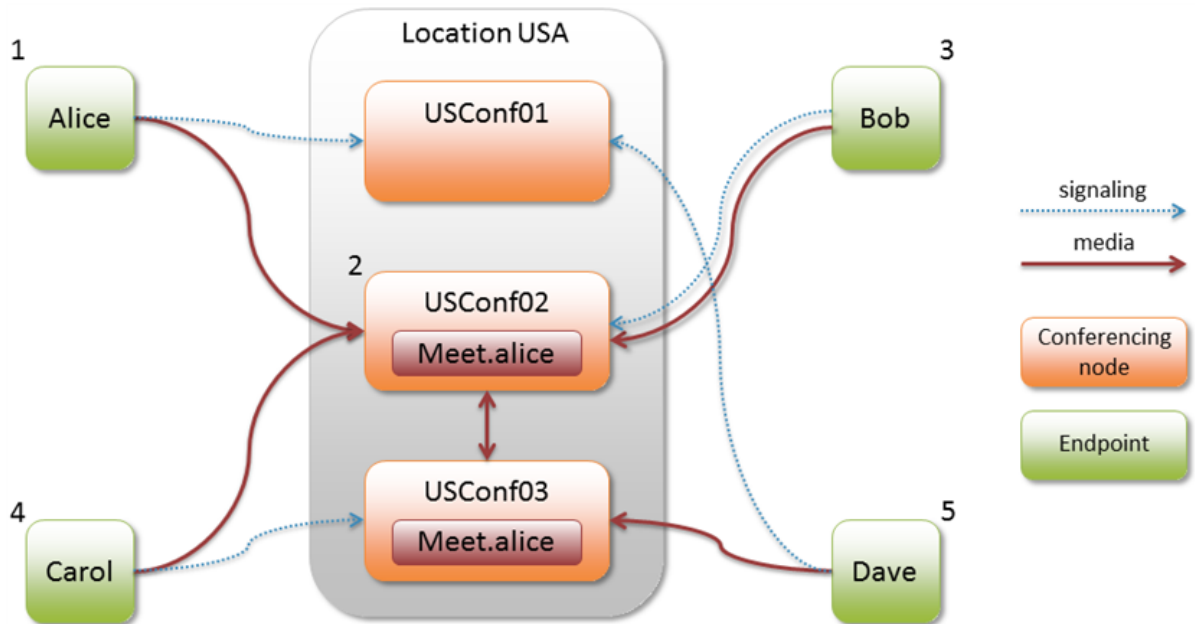
Based on these rules, here are some examples of how signaling and media are routed in different situations:

- [Media and signaling flows in a locally distributed conference](#)
- [Media and signaling flows in a globally distributed conference](#)

- [Media and signaling flows when using Proxying Edge Nodes and Transcoding Conferencing Nodes](#)

Media and signaling flows in a locally distributed conference

This example shows the signaling and media flows for a locally distributed conference (the conference is hosted on nodes all within the same location). In this example, all of the Conferencing Nodes involved are Transcoding Conferencing Nodes.



1. Alice dials `meet.alice@example.com` and is connected via DNS to Conferencing Node **USConf01**.
2. Pexip Infinity determines that another Conferencing Node within that location, **USConf02**, has the most available capacity and so sets up the conference on that node. Alice's signaling is handled by **USConf01** and the media is being sent directly to **USConf02**.
3. Bob dials `meet.alice@example.com`. He is connected to **USConf02**. The conference is already running on that Conferencing Node, so **USConf02** handles both the media and the signaling for Bob.
4. Carol dials `meet.alice@example.com`. She is connected to **USConf03**. Her signaling is handled by **USConf03** but the media is routed directly to **USConf02**, where the conference is running.
5. Dave dials `meet.alice@example.com`. He is connected to **USConf01**. Pexip Infinity determines that **USConf02** is now out of capacity, and **USConf03** has the most available capacity. Dave's media is routed to **USConf03**, but **USConf01** continues to handle his signaling.

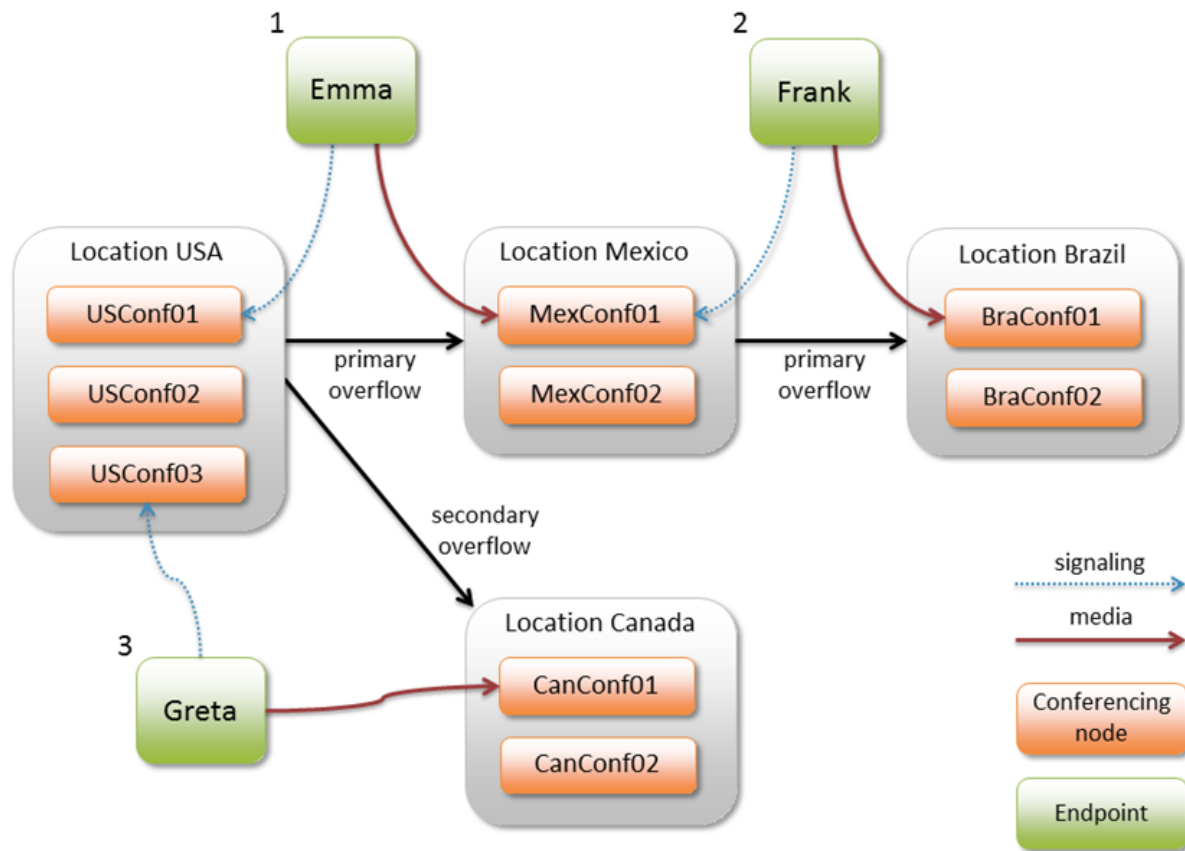
Media and signaling flows in a globally distributed conference

This example builds on the previous example described above. It assumes that the conference `meet.alice@example.com` is in progress and currently has all of its media and signaling in a locally distributed conference in location **USA**. Again, in this example, all of the Conferencing Nodes involved are Transcoding Conferencing Nodes.

This deployment has locations configured for USA, Mexico, Canada and Brazil, and the USA and Mexico locations are configured with transcoding, primary and secondary overflow locations as follows:

Location	Transcoding location	Primary overflow location	Secondary overflow location
USA	<i>This location</i> i.e. USA	Mexico	Canada
Mexico	<i>This location</i> i.e. Mexico	Brazil	-

This example now shows what could happen when three new participants, two from the USA and one from Mexico, join the conference.



1. Emma, who lives in USA, dials `meet.alice@example.com` and is connected to **USConf01** in location **USA**. Assume that all three Conferencing Nodes in location **USA** have reached their capacity, so Emma's media is routed directly to **MexConf01** in location **Mexico** (because **Mexico** is the primary overflow location for the **USA** location). **USConf01** continues to handle her signaling.
2. Frank, who lives in Mexico, dials `meet.alice@example.com` and is connected to **MexConf01** in location **Mexico**. Assume that all of the Conferencing Nodes in location **Mexico** have reached their capacity, so Frank's media is routed directly to **BraConf01** in location **Brazil** (because **Brazil** is the primary overflow location for the **Mexico** location to which Frank is connected). **MexConf01** continues to handle his signaling.
3. Greta, who lives in USA, dials `meet.alice@example.com` and is connected to **USConf03** in location **USA**. Assume that all of the Conferencing Nodes in location **USA** have reached their capacity, as have all of the nodes in the USA location's primary overflow location of **Mexico**, so Greta's media is routed directly to **CanConf01** in location **Canada** (because **Canada** is the secondary overflow location for the **USA** location to which Greta is connected; it is irrelevant if location **Brazil** has spare capacity as **Brazil** is not an overflow location for endpoints connected to the **USA** location). **USConf03** continues to handle her signaling.

Note that this example deliberately describes an extreme scenario as its purpose is to demonstrate how location overflow logic is applied.

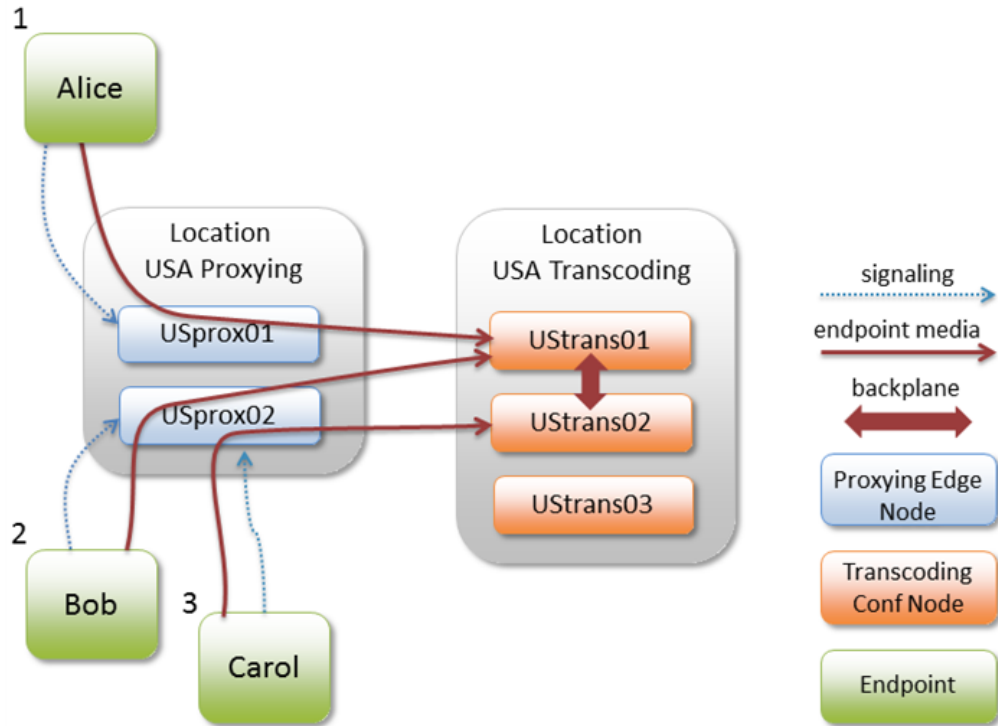
Conference escalation from locally to globally distributed is handled automatically by Pexip Infinity and is seamless to conference participants.

Media and signaling flows when using Proxying Edge Nodes and Transcoding Conferencing Nodes

These examples show the signaling and media flows when you have Proxying Edge Nodes in your deployment. Here, we assume that all callers located in the USA are routed via DNS to a Proxying Edge Node in the **USA Proxying** location, and that location is configured with a Transcoding location of **USA Transcoding** as follows:

Location	Transcoding location	Primary overflow location	Secondary overflow location
USA Proxying	USA Transcoding	Mexico Transcoding	-
Canada Proxying	Canada Transcoding	Mexico Transcoding	-

1. Alice dials



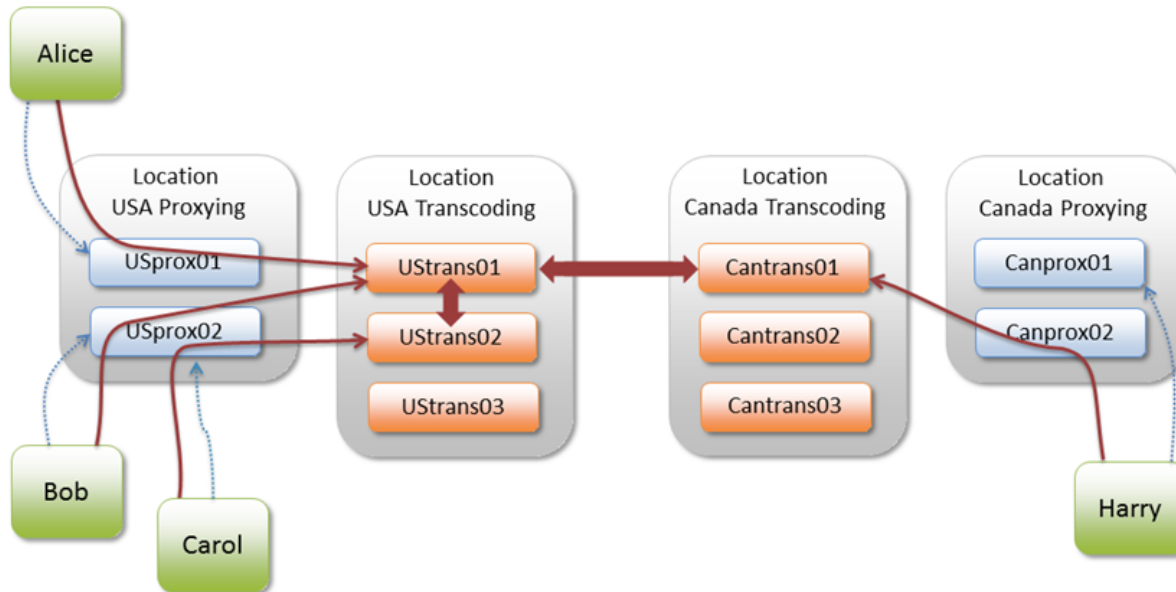
meet.alice@exam

ple.com and is connected via DNS to Proxying Edge Node **USprox01**. Her signaling remains routed to **USprox01** throughout the call. **USprox01** is also selected as the proxying node to handle Alice's media, from where it needs to be proxied to a transcoding resource. Pexip Infinity chooses the node with the most available capacity in the transcoding location of **USA Transcoding** — in this case **UStrans01**. Therefore **USprox01** forwards Alice's media onto **UStrans01** which hosts the conference meet.alice.

- Bob dials `meet.alice@example.com` and he is connected to **USprox02** for signaling purposes. **USprox02** is also chosen to handle his media (as it is the proxying node in that location with the most available capacity) and it needs to proxy his media onto a transcoding resource in the transcoding location. The meet.alice conference is already running on **UStrans01** and has spare capacity, so **USprox02** forwards Bob's media onto **UStrans01**.
- Carol dials `meet.alice@example.com` and she is connected to **USprox02**. Her media is also allocated to **USprox02**. However, this time **UStrans01** has no spare capacity, so Pexip Infinity chooses another node in the transcoding location as the forwarding destination — **UStrans02** in this case. The conference is now locally distributed within the **USA Transcoding** location and a local backplane is established between **UStrans01** and **UStrans02**.

As before, if the nodes in the transcoding location reach their capacity, the media would be forwarded instead onto any overflow locations that are configured against the location handling the signaling, which is **Mexico Transcoding** in this example.

Escalated to a globally distributed conference



We can now extend this example to show how the conference could become globally distributed when a participant in Canada joins. Here, the **Canada Proxying** location contains Proxying Edge Nodes and it is configured with a transcoding location of **Canada Transcoding**.

Harry dials `meet.alice@example.com` and he is routed via DNS to his local Proxying Edge Node **Canprox01**. His media proxying is allocated to **Canprox02** from where it is forwarded to **Cantrans01** in the transcoding location. The `meet.alice` conference is now globally distributed and therefore a geo backplane is set up between one of the nodes in location **USA Transcoding** (**UStrans01** in this example) and **Cantrans01**.

Implementing a dial plan

When you deploy the Pexip Infinity platform in your network, you must consider your dial plan and ensure that calls are routed appropriately. This includes:

- aliases that are associated with a conferencing service (Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service)
- person-to-person calls handled by the Infinity Gateway
- calls routed via the Infinity Gateway into an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.

i If your environment includes a PSTN gateway or uses an ITSP (Internet telephony service provider), consider the potential for toll fraud if you have Call Routing Rules that can route calls to the PSTN gateway or ITSP, or if you allow conference participants to dial out to other participants via the PSTN gateway or ITSP. See [PSTN gateways and toll fraud](#) for more information.

Service precedence

Incoming calls received by Pexip Infinity are routed as follows:

1. Pexip Infinity receives an incoming call via one of its Conferencing Nodes.
2. It checks whether the destination alias belongs to a Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service; if so, it directs the call to that service.
3. If the alias does not belong to any of the above services, Pexip Infinity checks the Call Routing Rules to see if the alias matches any rules specified there for incoming calls. If so, it places an Infinity Gateway call to the destination alias according to the rule's call target settings (which protocol, location and call control system to use, whether to route to registered devices only, etc).

This means that if an alias matches both a Virtual Meeting Room and a Call Routing Rule, the former will always take precedence and the call will be routed to the Virtual Meeting Room. You must therefore ensure that any regular expressions used in a Call Routing Rule

do not unintentionally overlap with any aliases used by a Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service.

Pexip Infinity's call routing logic i.e. the order in which calls are processed, is shown in the following diagram:



* This information can alternatively be provided by using external or local policy.

† If it is a non-Regular type of Virtual Reception (i.e. Google Meet, Microsoft Teams or SFB/Lync) the Virtual Reception captures the meeting code for the externally-hosted conference, which then needs to be matched by a suitable Call Routing Rule.

For more information about incoming and outgoing Call Routing Rules and how to specify where calls are routed, see [Configuring Call Routing Rules](#).

Using a standard format for aliases

We recommend that you use a standard format for all Virtual Meeting Room and Virtual Auditorium aliases within your Pexip Infinity deployment. A simple way to do this would be to have a set prefix such as `meet.`, followed by a user name, followed by your domain. For example:

- `meet.alice.smith@example.com` (for Alice's VMR)
- `meet.bob.jones@example.com` (for Bob's VMR)
- `meet.sales@example.com` (for the sales team's Virtual Auditorium)
- `meet.reception@example.com` (for the Virtual Reception)

Using this standard format will then make it simple to configure your call management system to route calls matching the format to the appropriate Conferencing Nodes.

Routing calls to the local Conferencing Node

When your call management system receives a call to a Virtual Meeting Room, Virtual Auditorium or Virtual Reception alias, it must then decide where to route the call.

A single conference can be hosted across one, two or more Conferencing Nodes with no difference in conference experience from the users' perspective. We recommend that in the first instance, calls are routed to a Conferencing Node that is geographically nearest to the endpoint that placed the call. This reduces the WAN bandwidth used for the conference in two ways: firstly, endpoints are only making local calls; and secondly, if the same conference is hosted in more than one location, only one set of media has to be sent between those locations.

We recommend that you set up rules on your call management system so that calls originating from a particular location (defined by zone, subzone or IP address range) and placed to a Pexip Infinity alias are routed to a specific Conferencing Node. If you are deploying Proxying Edge Nodes, we recommend that you ensure that calls are routed only to those Proxying Edge Nodes and not to Transcoding Conferencing Nodes. To build on the previous example, we have configured subzones on our call management system which group together endpoints in a particular physical location, and we have then set up the following call routing rules:

Source	Alias	Destination zone
Oslo subzone	<code>meet.*@example.com.*</code>	Norway Conferencing Node
New York subzone	<code>meet.*@example.com.*</code>	Americas Conferencing Node
Boston subzone	<code>meet.*@example.com.*</code>	Americas Conferencing Node

In this case, users in Oslo, New York and Boston could all call `meet.sales@example.com` and be routed to the same conference.

See [Regular expression \(regex\) reference](#) for information about writing regular expressions.

Further information

For further information about how to configure your specific call management system to work with Pexip Infinity, see the following documentation:

- [Pexip Infinity and Microsoft Skype for Business / Lync Deployment Guide](#)
- [Pexip Infinity and Cisco VCS Deployment Guide](#)
- [Pexip Infinity and Cisco Unified Communications Manager Deployment Guide](#)
- [Pexip Infinity and Polycom DMA Deployment Guide](#)

DNS record examples

This section describes the DNS records required to allow endpoints and clients to access Pexip Infinity services.

You can use the tool at <http://dns.pexip.com> to lookup and check SRV records for a domain.

Enabling endpoints to discover Conferencing Nodes

DNS A-records are required for endpoints and clients to discover Pexip Infinity Conferencing Nodes. Separate, additional records that resolve to these DNS A-records are required for [SIP and H.323 endpoints](#), [federated Skype for Business / Lync environments](#) and [Infinity Connect clients](#) respectively.

The examples below show DNS A and SRV records for the domain **vc.example.com** to allow:

- SIP and H.323 endpoint registration messages to be routed to Pexip Infinity
- calls from SIP and H.323 endpoints to be routed to Pexip Infinity
- calls from remote Skype for Business / Lync environments to Pexip Infinity
- Poly/Polycom endpoints to register to One-Touch Join
- Infinity Connect desktop client and mobile clients for Android to register to Pexip Infinity
- Infinity Connect clients to make calls to Pexip Infinity.

If you have multiple Conferencing Nodes, you must set up DNS A and SRV records for each node.

Host DNS A-records

This example assumes there are 2 Conferencing Nodes, thus 2 DNS A-records are required.

Hostname	Host IP address
px01.vc.example.com.	198.51.100.40
px02.vc.example.com.	198.51.100.41

In your actual deployment, both the **Hostname** and **Host IP address** should be changed to use the real names and addresses of your Conferencing Nodes.

DNS SRV records for SIP and H.323 endpoints

The following DNS SRV records are required for H.323 and SIP services. For each service there should be one record per host (Conferencing Node). Therefore, in this example, there are 2 records per service — one for each target host — **px01.vc.example.com** and **px02.vc.example.com**.

Name	Service	Protocol	Priority	Weight	Port	Target host
vc.example.com.	h323cs	tcp	10	10	1720	px01.vc.example.com.
vc.example.com.	h323cs	tcp	10	10	1720	px02.vc.example.com.
vc.example.com.	h323ls	udp	10	10	1719	px01.vc.example.com.
vc.example.com.	h323ls	udp	10	10	1719	px02.vc.example.com.
vc.example.com.	h323rs	udp	10	10	1719	px01.vc.example.com.
vc.example.com.	h323rs	udp	10	10	1719	px02.vc.example.com.
vc.example.com.	sip	tcp	10	10	5060	px01.vc.example.com.
vc.example.com.	sip	tcp	10	10	5060	px02.vc.example.com.
vc.example.com.	sips	tcp	10	10	5061	px01.vc.example.com.
vc.example.com.	sips	tcp	10	10	5061	px02.vc.example.com.
vc.example.com.	sip	udp *	10	10	5060	px01.vc.example.com.
vc.example.com.	sip	udp *	10	10	5060	px02.vc.example.com.

* Only required if you intend to enable SIP over UDP.

In your actual deployment, both the **Name** and the **Target host** should be changed to use the real domain name and host names of your Conferencing Nodes.

If you have a mixture of public-facing and privately-addressed Conferencing Nodes, ensure that your public-facing SRV records refer to your public-facing Conferencing Nodes (A-records), and vice versa for your local DNS records and your privately-addressed Conferencing Nodes.

In these examples, the DNS records would be:

```
_h323cs._tcp.vc.example.com. 86400 IN SRV 10 10 1720 px01.vc.example.com.
_h323cs._tcp.vc.example.com. 86400 IN SRV 10 10 1720 px02.vc.example.com.

_h323ls._udp.vc.example.com. 86400 IN SRV 10 10 1719 px01.vc.example.com.
_h323ls._udp.vc.example.com. 86400 IN SRV 10 10 1719 px02.vc.example.com.

_h323rs._udp.vc.example.com. 86400 IN SRV 10 10 1719 px01.vc.example.com.
_h323rs._udp.vc.example.com. 86400 IN SRV 10 10 1719 px02.vc.example.com.

_sip._tcp.vc.example.com.      86400 IN SRV 10 10 5060 px01.vc.example.com.
_sip._tcp.vc.example.com.      86400 IN SRV 10 10 5060 px02.vc.example.com.

_sips._tcp.vc.example.com.     86400 IN SRV 10 10 5061 px01.vc.example.com.
_sips._tcp.vc.example.com.     86400 IN SRV 10 10 5061 px02.vc.example.com.

_sip._udp.vc.example.com.      86400 IN SRV 10 10 5060 px01.vc.example.com.
_sip._udp.vc.example.com.      86400 IN SRV 10 10 5060 px02.vc.example.com.

px01.vc.example.com.           86400 IN A 198.51.100.40
px02.vc.example.com.           86400 IN A 198.51.100.41
```

i If the ability to implement intelligent DNS (GeoDNS, GSLB, etc.) exists, then different weights and priorities could be set on the SRV records for different locations. For example, if there is both a European and a North American site, each with separate DNS servers, the European one could return European Pexip nodes at a higher priority than the North American nodes, and vice versa.

Enabling direct federation with remote Skype for Business / Lync environments

In public DMZ / hybrid deployments with remote Skype for Business / Lync environments, to ensure that calls from those remote Sfb/Lync environments are routed to your Conferencing Nodes, a DNS SRV record in the format **_sipfederationtls._tcp.<domain>** must be created.

Example

Assume that the following **_sipfederationtls._tcp.vc.example.com** DNS SRV and associated A-records have been created:

```
_sipfederationtls._tcp.vc.example.com. 86400 IN SRV 1 100 5061 px.vc.example.com.
px.vc.example.com.                     86400 IN A 198.51.100.40
px.vc.example.com.                     86400 IN A 198.51.100.41
```

The SRV record points to the DNS A-record **px.vc.example.com**, port 5061, with a priority of 1 and a weight of 100. In other words, it tells Skype for Business / Lync to send its TLS requests to **px.vc.example.com** on TCP port 5061.

We then have 2 round-robin DNS A-records for the **px.vc.example.com** pool hostname that resolves **px.vc.example.com** to the IP addresses of our 2 Conferencing Nodes (198.51.100.40 and 198.51.100.41).

Even if you only intend initially to use a single Conferencing Node to receive incoming Sfb/Lync calls, this pool-based approach allows you to easily add more nodes in the future. (In your actual deployment, the hostnames (including the pool hostname) and host IP addresses should be changed to use the real hostnames and IP addresses of your Conferencing Nodes.)

Note that these A-records specified for the **px.vc.example.com** pool are required in addition to the "standard" A-records that will exist for each Conferencing Node based on their individual hostnames and resolve to the same IP addresses, as shown in the Host DNS A records section above.

The domain name used in the **_sipfederationtls._tcp.<domain>** SRV record has to match the domain in the corresponding A-record. This is required due to the trust model for Sfb/Lync federation. For example:

- An SRV record such as **_sipfederationtls._tcp.vc.example.com** must have a corresponding A-record with the same domain, such as **px.vc.example.com**.
- You cannot, for example, configure the **_sipfederationtls._tcp.vc.example.com** SRV record to point to **px.video.example.com** or **px01.otherdomain.com**.

Enabling Poly/Polycom endpoints to register to One-Touch Join

If you have a One-Touch Join deployment that includes Poly endpoints in a location with more than one Conferencing Node, you should spread the Poly endpoint registrations across all nodes in the location to maximize performance and provide redundancy. To achieve this, we recommend that all Poly endpoints in a location register to a single FQDN which uses round-robin DNS to resolve to each Conferencing Node in turn. This will require you to set up appropriate DNS records for all Conferencing Nodes in the location, and ensure that your DNS server is configured to round-robin between these records.

These records are specific to One-Touch Join, and are required in addition to the "standard" A-records that will exist for each Conferencing Node (which are based on their individual hostnames and resolve to the same IP addresses).

Note that all Conferencing Nodes in the location that is being used for One-Touch Join must be included in the DNS records.

Example

In this example, we have two locations being used for One-Touch Join that contain Poly endpoints. Each location has three Conferencing Nodes. We set up the following DNS A-records:

```
otj01.vc.example.com.      86400 IN A 10.44.0.10
otj01.vc.example.com.      86400 IN A 10.44.0.11
otj01.vc.example.com.      86400 IN A 10.44.0.12
otj02.vc.example.com.      86400 IN A 10.44.0.20
otj02.vc.example.com.      86400 IN A 10.44.0.21
otj02.vc.example.com.      86400 IN A 10.44.0.22
```

All Poly endpoints in the first location are configured with **otj01.vc.example.com** as their Exchange Server, and those in the second location are configured with **otj02.vc.example.com**.

Enabling access from the Infinity Connect mobile client and the Infinity Connect desktop client

You must set up DNS records so that the Infinity Connect clients know which host to contact when placing calls or registering to Pexip Infinity.

The host will typically be a public-facing Conferencing Node (for on-premises deployments where your Transcoding Conferencing Nodes are located within a private network we recommend that you deploy public-facing Proxying Edge Nodes).

To enable access from the Infinity Connect desktop clients and Infinity Connect mobile clients, each domain used in aliases in your deployment must either have a DNS SRV record for **_pexapp._tcp.<domain>**, or resolve directly to the IP address of a public-facing Conferencing Node.

The SRV records for **_pexapp._tcp.<domain>** should always:

- point to an FQDN which **must** be valid for the TLS certificate on the target Conferencing Nodes
- reference port 443 on the host.

Example

Assume that the following **_pexapp._tcp.vc.example.com** DNS SRV records have been created:

```
_pexapp._tcp.vc.example.com. 86400 IN SRV 10 100 443 px01.vc.example.com.
_pexapp._tcp.vc.example.com. 86400 IN SRV 20 100 443 px02.vc.example.com.
```

These point to the DNS A-records **px01.vc.example.com**, port 443 (HTTPS), with a priority of 10 and a weight of 100, and **px02.vc.example.com**, port 443, with a relatively lower priority of 20 and a weight of 100.

This tells the Infinity Connect desktop and mobile clients to initially send their HTTP requests to host **px01.vc.example.com** (our primary node) on TCP port 443. The clients will also try to use host **px02.vc.example.com** (our fallback node) if they cannot contact px01.


Testing and next steps after initial installation

After you have completed your installation and initial configuration of Pexip Infinity, you can make a test call to check that your system is working. You can also extend your deployment by integrating it with other call control or third-party systems, or by customizing the user experience. You should also consider how to let your users know about their new video conferencing service.

Making a test call

When you have deployed a Conferencing Node and configured a Virtual Meeting Room and an alias, you can make a test call to check that your system is working.

An easy way to do this is by using the Infinity Connect web app to dial the alias of one of the Virtual Meeting Rooms you've already created, as follows:

1. Open a browser (we recommend Chrome) and type in the IP address (or FQDN, if you've set it up already) of one of the Conferencing Nodes.
2. Ensure that your camera and microphone are enabled and working correctly:
 - You should see your own image in the video window.
 - The microphone icon should be green  and you should see a green bar under the video image indicating the volume of audio being detected.



3. Select **video**.
4. In the **Search to call** field, enter the alias of the VMR you want to use for testing and then press **Enter**.
You will be connected to the VMR.
5. From another device, join the conference in the same way.

The two participants should be able to see and hear each other, and share content.

See [About the Infinity Connect web app](#) for more information.

Further configuration

You are now ready to continue [configuring the Pexip Infinity platform](#) and [services](#) and deploying more [Conferencing Nodes](#).

Specifically, you should now do the following:

- [Assigning hostnames and FQDNs](#)
- [Monitoring via SNMP](#)

At some point you may also want to:

- [integrate the Pexip Infinity platform with your call control system](#)
- [configure the Pexip Infinity Distributed Gateway](#)
- [register devices directly to the Pexip Infinity platform](#)
- [customize the user experience](#)

Integrating with a call control system

To integrate Pexip Infinity with your call control system, you must configure a trunk or neighbor zone towards each of the Conferencing Nodes.

For further information about how to configure your specific call management system to work with Pexip Infinity, see the following documentation:

- [Pexip Infinity and Microsoft Skype for Business / Lync Deployment Guide](#)
- [Pexip Infinity and Cisco VCS Deployment Guide](#)
- [Pexip Infinity and Cisco Unified Communications Manager Deployment Guide](#)
- [Pexip Infinity and Polycom DMA Deployment Guide](#)

Configuring the Pexip Infinity Distributed Gateway

The Pexip Infinity Distributed Gateway ("Infinity Gateway") enables endpoints to make calls to other endpoint devices or systems. This includes calls between devices that use different protocols and media formats, such as SIP and H.323 systems, Skype for Business clients (MS-SIP), and Infinity Connect clients (WebRTC). It also enables you to route calls from VTCs and standards-based endpoints into an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.

Registering devices directly to the Pexip Infinity platform

SIP and H.323 endpoints, and some Infinity Connect clients can register directly to Pexip Infinity Conferencing Nodes. This allows Pexip Infinity to route outbound calls to those registered devices without having to go via a SIP proxy or H.323 gatekeeper, or rely on DNS.

Customizing the user experience

You can easily apply your own corporate branding to the Pexip Infinity platform, and produce a personalized user experience for all of your Pexip Infinity services.

Informing users about the new video conferencing service

Finally, you'll need to let your end users know about the new video conferencing services available to them, and how they can use it. The following end user guides are available:

- [Using your Virtual Meeting Room](#)
- [Using the Infinity Connect web app](#)
- [Using the Infinity Connect desktop client](#)
- [Using the Infinity Connect mobile client](#)

We also have provided some [Example emails for sending to new users](#), which you can use as a basis for the information you provide to your users.

Administering Pexip Infinity

For information about how to administer Pexip Infinity, see:

Using the Pexip Infinity Administrator interface	93
Pexip Infinity system configuration	95
Pexip Infinity platform configuration	105
Conferencing Node configuration	156
Pexip Infinity conference types	180
Pexip Infinity conference settings	228
Controlling active conferences	260
Customizing with themes	269
Integrating Google Meet with Pexip Infinity	299
Integrating Microsoft Teams with Pexip Infinity	318
Integrating Epic telehealth with Pexip Infinity	319
Integrating Pexip Infinity with authentication and provisioning services	339
Pexip Infinity maintenance tasks	387
Best practices	413

Using the Pexip Infinity Administrator interface

After you have run the installation wizard, all further configuration should be done using the Pexip Infinity Administrator interface.

This topic covers:

- [Accessing the Pexip Infinity Administrator interface](#)
- [Setting the session timeout](#)
- [Changing the display language](#)
- [Timezones](#)
- [Getting help and support](#)

Accessing the Pexip Infinity Administrator interface

The Pexip Infinity Administrator interface is hosted on the Management Node. To access this:

1. Open a web browser and type in the IP address or DNS name that you assigned to the Management Node using the installation wizard (you may need to wait a minute or so after installation is complete before you can access the Administrator interface).
2. Until you have uploaded appropriate TLS certificates to the Management Node, your browser may present you with a warning that the website's security certificate is not trusted. You should proceed, but upload appropriate TLS certificates to the Management Node (and Conferencing Nodes, when they have been created) as soon as possible.

The **Pexip Infinity Conferencing Platform** login page will appear.

3. Log in using the web administration username and password you set using the installation wizard.

You are now ready to begin configuring the Pexip Infinity platform and deploying Conferencing Nodes.

As a first step, we strongly recommend that you configure at least 2 additional NTP servers or NTP server pools to ensure that log entries from all nodes are properly synchronized.

It may take some time for any configuration changes to take effect across the Conferencing Nodes. In typical deployments, configuration replication is performed approximately once per minute. However, in very large deployments (more than 60 Conferencing Nodes), configuration replication intervals are extended, and it may take longer for configuration changes to be applied to all Conferencing Nodes (the administrator log shows when each node has been updated).

Setting the session timeout

By default, Administrator interface web sessions time out after 30 minutes of inactivity. You can modify the **Management web interface session timeout** setting via the [Global Settings](#) page (Platform > Global Settings > Management Web Interface Configuration). You can also disable timeouts completely by deselecting the **Enable management web interface session timeout** setting.

Changing the display language


The Administrator interface supports localization, meaning it can be displayed in different languages. It includes the following languages:

- English (default)
- Simplified Chinese
- Russian


To select the language, you must edit the language preferences within your browser settings. You must then refresh your browser for the changes to take effect.

You can edit your browser's language preferences as follows:

Chrome

1. Select the control button  at the top right of the screen.
2. Select **Settings**.
3. Select **Show advanced settings**.
4. In the **Languages** section, select **Language and input settings....**

Firefox

1. Select the Firefox button  at the top left of the screen.
2. Select **Options > Options**.
3. Select the **Content** tab and in the **Languages** section, select **Choose....**

Timezones

The Administrator interface displays all times (except for those in the [Administrator log](#) and [Support log](#), and [usage graphs](#)) in your local time. This is obtained from the timezone configured on the computer you are using to access the Administrator interface. So for example, if you are sitting in an office in London and you log in to a Management Node that is located in New York, call start and end times will be shown in London time.


The timezone being used is shown in brackets after the time, in the format **2019-02-25 22:00:12 (GMT)**.

Logs (and usage graphs, which are based on the information in the logs) are always shown in UTC because of the distributed nature of the Pexip Infinity platform. Logs use the format **2019-02-24T23:16:33.019+00:00**.

Note that the VMs hosting the Management Node and Conferencing Nodes use the UTC timezone. Do not attempt to change the timezone on these systems.

Getting help and support

There are two types of help integrated into the Pexip Infinity Administrator interface:

- **Field-level help:** by default, most configuration fields show an explanation of the field underneath the input area. You can turn these explanations on or off by clicking on the  icon on the bottom right of the page.
- **Context-sensitive help:** from any page, clicking on the **Help** link at the top right of the page opens a new browser window showing relevant information from the **Pexip Infinity Administrator Guide**. From there you can use the table of contents on the left-hand side of the help window to navigate and view the entire guide. There is also a search box at the top right of the browser window which you can use to search for individual words or phrases.

More deployment information and PDF downloads of all documentation are available on the [Pexip Infinity technical documentation website](#).

If you cannot find the information you require, contact your Pexip authorized support representative. Technical support for software issues is available while under a valid support contract and running a Pexip Infinity version no more than 2 major software releases behind the current release. Software bug fixes will only be provided in either the current or the next major release of software.

For an in-depth understanding of the Pexip Infinity platform, we recommend that you attend an appropriate training course — for more information, visit the [Pexip Academy](#).

Pexip Infinity system configuration

Configuring DNS servers

Pexip Infinity uses DNS to resolve the hostnames of external system components including NTP servers, syslog servers, SNMP servers and web proxies. It is also used for call routing purposes — SIP proxies, gatekeepers, external call control and conferencing systems and so on. The address of at least one DNS server must be added to your system.

To add, edit or delete the set of available DNS servers, go to **System > DNS Servers**.

The available options are:

Option	Description
Address	The IP address of the DNS server.
	The address must be an IPv4 address. (Note that IPv6 DNS resolution does not require an IPv6-addressed DNS server.)
Description	An optional description of the DNS server.

After configuring the DNS servers available to your system, you should assign appropriate DNS servers to each location (**Platform > Locations**). Each Conferencing Node in that location will then use those DNS servers.

You can also select the specific DNS servers to be used by the Management Node (**Platform > Management Node**).

- i** While you can assign unlimited DNS servers to a location / Management Node, only three will be used. They are used in the order in which they were assigned to the location / Management Node, with the first to be assigned having highest priority. If multiple servers are assigned simultaneously, those servers are used in descending numerical order. Hence, the order in which the DNS servers are prioritized is not necessarily the same as the order in which they are displayed. The order in which they were configured via **System > DNS Servers** is irrelevant.

Syncing with NTP servers

Pexip Infinity uses NTP servers to obtain accurate system time. This is necessary to ensure correct operation, including configuration replication and log timestamps.

- i** All host servers **must** be synchronized with accurate time before you install the Management Node or Conferencing Nodes on them.
- i** NTP **must** be enabled on the Management Node VM before you deploy any Conferencing Nodes (this is done during installation of the Management Node).

We strongly recommend that you configure at least three distinct NTP servers or NTP server pools on all your host servers and the Management Node itself. This ensures that log entries from all nodes are properly synchronized.

The VMs hosting the Management Node and Conferencing Nodes use the UTC timezone, and all logs are in UTC. Do not attempt to change the timezone on these systems. Note however that the administrator web interface uses your local time.

To add, edit or delete the set of available NTP servers, go to **System > NTP Servers**.

The available options are:

Option	Description
Address	The IP address or FQDN of the NTP server.
Description	An optional description of the NTP server.
Secure NTP key ID	This optional field allows you to configure a key ID which is used in conjunction with the Secure NTP key for authenticating access to a secure NTP server.
Secure NTP key	This optional field, used in conjunction with the Secure NTP key ID, allows configuration of a SHA1 key for authenticating access to a secure NTP server. Enter the plaintext key; this will be stored as a SHA1 hash.

After configuring the NTP servers available to your system, you should assign appropriate NTP servers to each location (**Platform > Locations**). Each Conferencing Node in that location will then use those NTP servers.

You can also select the specific NTP servers to be used by the Management Node (**Platform > Management Node**).

Using a web proxy

You can specify one or more web proxies to use for outbound web requests from the Management Node, and from the Conferencing Nodes in each system location. Web proxies, when configured, are used to route all requests for analytics and incident reporting, Epic telehealth requests, some requests for license activation, requests from the Management Node to cloud service providers, and One-Touch Join requests. Support for other types of outbound web requests will be added in later releases.

When a web proxy is configured for the **Management Node**, it is used to route outbound web requests in the following situations:

- when sending [usage statistics](#) and [incident reports](#),
- for [license activation](#) requests sent to [activation.pexip.com](#),
- when the Management Node is communicating with the service configured for [cloud bursting](#) (i.e. Azure, AWS or GCP), and
- (for deployments where One-Touch Join has been enabled and is using OAuth) when the Management Node is sending requests to the OAuth token endpoint.

When a web proxy is configured for a **system location**:

- all Conferencing Nodes in that location will use that web proxy for [incident reporting](#), [Epic telehealth REST API requests](#), and
- if the system location is being used for One-Touch Join, all One-Touch Join-related outbound requests from Conferencing Nodes in that location will use the web proxy. These requests include:
 - connections to Cisco endpoints
 - connections to the Exchange server
 - connections to Google Workspace
 - (for deployments where One-Touch Join is using OAuth) for sending requests to the OAuth token endpoint.

To use a web proxy, you must first add its details via **System > Web Proxies**, and then add it to the configuration for the Management Node (**Platform > Management Node**) or the system location (**Platform > Locations**) you wish to use it for.

Option	Description
Name	The name used to refer to this proxy server.
Address	The IP address or FQDN of the proxy server.
Port	The IP port of the proxy server. Default: 8080.
Username	The username and password used when accessing the proxy server.
Password	

Next steps

You must now add the web proxy to the [Management Node](#) or to one or more [System Locations](#).

Monitoring via SNMP

The Pexip Infinity Management Node and Conferencing Nodes can be monitored using SNMP, and they can also be configured to send SNMP traps to an SNMP Network Management System (NMS). This topic covers:

- [SNMP MIB access](#)
- [Sending trap notifications to an NMS](#)
- [Configuring an SNMP NMS](#)
- [Configuring SNMP monitoring on the Management Node](#)
- [Configuring SNMP monitoring on Conferencing Nodes](#)

SNMP MIB access

Pexip Infinity supports SNMPv2c (non-secure) and SNMPv3 (secure) access to the basic RFC 1213 MIB-II tree (1.3.6.1.2.1) with read-only functionality. This includes full or partial support for:

- system_mib
- interfaces
- snmp_mib
- at
- ip
- icmp
- udp
- tcp
- RFC 1514/RFC 2790 MIB-II host MIB

Sending trap notifications to an NMS

In addition to enabling read access to the MIB tree, you can also nominate an SNMP Network Management System (NMS) to receive trap notifications from the Management Node and from the Conferencing Nodes within a system location.

The supported SNMP traps are:

Trap	OID	Description
cold start	1.3.6.1.6.3.1.1.5.1	Emitted when the snmpd service running on the node starts or restarts (due to snmp being reconfigured and/or due to the Conferencing Node rebooting).
authentication failure	1.3.6.1.6.3.1.1.5.5	Generated, for example, when any attempt to query SNMP values is made using an incorrect community string.
warm start	1.3.6.1.6.3.1.1.5.2	Generated when any software component fails unexpectedly (and coincides with the generation of a Pexip Incident Report).

i The SNMP support in Pexip Infinity is built on top of the popular **net-snmp** open source implementation and therefore inherits some of the same behaviors (for example, generating a coldstart rather than warmstart on reconfiguration). For this reason you may also see some **net-snmp**-specific traps, such as the **nsNotifyShutdown** trap (OID 1.3.6.1.4.1.8072.4.0.2) when the snmpd daemon shuts down.

Pexip Infinity does not currently support traps with SNMPv3. If traps are required, use SNMPv2c.

Configuring an SNMP NMS

If you want to send SNMP notifications from a Management Node or Conferencing Node to a SNMP NMS, you need to configure the details of the NMS. To do this:

1. Go to **System > SNMP NMSs** and select **Add SNMP Network Management System**.
2. Complete the required fields:

Option	Description
Name	The name used to refer to this NMS.
Description	An optional description of the NMS.
Address	The IP address or FQDN of the NMS.
Port	The SNMP port of the NMS. Default: 161.

Option	Description
SNMP trap community	The SNMP trap community name. Default: public

3. Select **Save**.


The NMS can now be selected when configuring the Management Node and system locations, as described below.

Configuring SNMP monitoring on the Management Node

To configure SNMP monitoring on the Management Node:

1. Go to **Platform > Management Node** and select the Management Node.
You are taken to the **Edit Management Node** page.
2. To enable SNMP read access for the Management Node, configure the following settings:

Option	Description
SNMP mode	Configures the SNMP access mode for the selected node: <i>Off</i> : SNMP is disabled. You cannot use SNMP to query the node for its status. <i>SNMPv2c read-only</i> : enables insecure, read-only access. <i>SNMPv3 read-only</i> : enables secure, read-only access, using the authPriv security level with SHA1 authentication and AES 128-bit encryption. When enabled, access is provided to the basic RFC 1213 MIB-II tree (1.3.6.1.2.1). Default: <i>Off</i> .
SNMP community	The SNMP group to which this node belongs. This setting applies to SNMPv2c only. Default: public
SNMPv3 username	The node's SNMPv3 username, used to authenticate SNMPv3 requests.
SNMPv3 privacy password	The node's SNMPv3 privacy password used for encrypting messages between the node and the management station. AES encryption must be used; DES is not supported.
SNMPv3 authentication password	The node's SNMPv3 authentication password, used to authenticate the associated username. The SHA authentication protocol must be used; MD5 is not supported.
SNMP system contact	The contact details (for example, email address) of the person responsible for this particular node.
SNMP system location	A description of the node's location.

3. If you want to send SNMP traps from the Management Node to an NMS, select the NMS from the **SNMP NMS** drop-down menu.
If you have not already added the NMS, you can do so now by clicking .
4. Select **Save**.


Configuring SNMP monitoring on Conferencing Nodes

To configure SNMP monitoring on a Conferencing Node:

1. Go to **Platform > Conferencing Nodes** and select the Conferencing Node.
You are taken to the **Edit Conferencing Node** page.

- To enable SNMP read access for the Conferencing Node, configure the following settings:

Option	Description
SNMP mode	<p>Configures the SNMP access mode for the selected node:</p> <p><i>Off</i>: SNMP is disabled. You cannot use SNMP to query the node for its status.</p> <p><i>SNMPv2c read-only</i>: enables insecure, read-only access.</p> <p><i>SNMPv3 read-only</i>: enables secure, read-only access, using the authPriv security level with SHA1 authentication and AES 128-bit encryption.</p> <p>When enabled, access is provided to the basic RFC 1213 MIB-II tree (1.3.6.1.2.1).</p> <p>Default: <i>Off</i>.</p>
SNMP community	<p>The SNMP group to which this node belongs. This setting applies to SNMPv2c only.</p> <p>Default: public</p>
SNMPv3 username	The node's SNMPv3 username, used to authenticate SNMPv3 requests.
SNMPv3 privacy password	<p>The node's SNMPv3 privacy password used for encrypting messages between the node and the management station.</p> <p>AES encryption must be used; DES is not supported.</p>
SNMPv3 authentication password	<p>The node's SNMPv3 authentication password, used to authenticate the associated username.</p> <p>The SHA authentication protocol must be used; MD5 is not supported.</p>
SNMP system contact	The contact details (for example, email address) of the person responsible for this particular node.
SNMP system location	A description of the node's location.

- If you want to send SNMP traps from the Conferencing Node to an SNMP NMS, you must ensure that the node's system location is configured with the NMS that will receive trap notifications from that node:
 - Go to **Platform > Locations**.
 - Select the Location to which the Conferencing Node belongs.
You are taken to the **Edit System Location** page.
 - From the **SNMP NMS** drop-down menu, select the NMS to which traps will be sent. Note that this NMS applies to all Conferencing Nodes in this location.
If you have not already added the SNMP NMS, you can do so now by clicking .
- Select **Save**.

Using a syslog server

The Management Node acts as a syslog server, collating the logs from itself and each Conferencing Node to produce the Pexip Infinity [support log](#). The support log includes the [administrator log](#).

However, you can also use a remote syslog server to collect copies of each system's logs. The advantage of this is that, should the Management Node become unavailable, logging would still continue on the syslog server.

When a remote syslog server is used, the Management Node and each of the Conferencing Nodes sends its logs directly to the syslog server using the syslog protocol. The Management Node will still continue to collate logs from all the nodes, and it is these unified logs that are shown from the Pexip Infinity Administrator interface. Note that the Management Node does not send the unified logs to the syslog server — it only sends its own logs.

The Management Node shows the **support log** only. If you elect to use a remote syslog server, you can choose to send it **audit logs** and **web server logs** in addition to, or instead of, the support log.

You can configure more than one remote syslog server, for example if you want to send different combinations of the support, audit and web server logs to different servers. If more than one syslog server is configured, logs will be sent to each of them.

To add, edit or delete the remote syslog servers used by Pexip Infinity, go to **System > Syslog Servers**.

The available options are:

Option	Description
Description	An optional description of the syslog server.
Address	The IP address or FQDN of the remote syslog server.
Port	The port on the remote syslog server. Default: 514.
Transport	The IP transport protocol used to communicate with the remote syslog server. Default: UDP.
Support log	Enables sending of support and administrator log entries to this syslog server.
Audit log	Enables sending of Linux audit log entries to this syslog server. Some security-related certifications may require this log to be recorded.
Web server log	Enables sending of web server log entries to this syslog server. This is a log of all HTTPS requests to the Management Node and Conferencing Nodes.

The following table shows the facility code used for each log type:

Facility code	Log type
local0	Admin log
local2	Support log
local6	Audit log
local7	Web server log

Web server logs are provided in the Apache web server format (http://httpd.apache.org/docs/current/mod/mod_log_config.html#formats) with the following fields:

Format string	Description
%h	Remote hostname
%l	Remote logname
%u	Remote user
%t	Time the request was received
%r	Received HTTP request
%P	The web server process ID
%s	Response status code
%b	Size of response in bytes
%D	The time taken to serve the request in microseconds <ul style="list-style-type: none">HTTP ReferrerHTTP User-Agent
%X	Connection status when response is completed

Configuring SMTP servers

SMTP servers are used in conjunction with [Sending provisioning emails to VMR and device owners](#) when bulk-provisioning VMRs and device aliases from Active Directory via LDAP.

To add, edit or delete the SMTP servers used by Pexip Infinity, go to **System > SMTP Servers**.


The available options are:

Option	Description
Name	The name used to refer to the SMTP server.
Description	A description of the SMTP server.
Address	The IP address or FQDN of the SMTP server.
Port	The port on the SMTP server to connect to. Default: 587.
Username	These are optional fields where you can specify the credentials of a valid account on the SMTP server.
Password	
From email address	The "from" email address to use when sending emails via this server. This must be an email address that is permitted to be used for sending email using this server and account.
Connection security	The type of connection security to use when connecting to this email server. Select <i>StartTLS</i> to use an encrypted connection. Default: None.

Managing static routes

Static routes are additional configuration settings that permit routing of traffic to networks that are not accessible through the default gateway.

Typically, static routes are configured on Conferencing Nodes that are deployed in a DMZ, and where the default gateway on those nodes routes traffic out to the internet. The static routes would allow those nodes to communicate with Pexip Infinity nodes or other systems in the local, internal network. (See [Network routing and addressing options for Conferencing Nodes](#) for more information.)

- You must configure the static routes you want to use (via **System > Static Routes**) before you can apply them to a Conferencing Node or a Management Node.
- Static routes can be assigned to Conferencing Nodes during the initial deployment process, as well as after it has been deployed.
 -  In some situations, for example where a Conferencing Node's default gateway is out to the public internet, a static route back to the Pexip Infinity platform must be applied to the Conferencing Node during its initial deployment phase (otherwise it will not be able to communicate with the Management Node and pick up its configuration).
- Static routes can only be assigned to a Management Node after it has been deployed.

The static routes take immediate effect after they have been assigned to a node. You do not have to restart the VM.

Configuring the set of available static routes

To configure the set of static routes that can be applied to a Conferencing Node or Management Node:

- Go to **System > Static Routes**.
- Select **Add Static route** and then configure the relevant destination and gateway addresses:

Option	Description
Name	A unique short name or description of the static route.

Destination network address	The IP address to be used in conjunction with the Network prefix to determine the network addresses to which this route applies.
Network prefix	<p>The prefix length used in conjunction with the Destination network address to determine the network addresses to which this route applies.</p> <p>The prefix length can be in the range 0–32 for IPv4 addresses and 0–128 for IPv6 addresses.</p> <p>For example, use a Destination network address of 10.0.0.0 and a Network prefix of 8 to route addresses in the range 10.0.0.0 to 10.255.255.255.</p>
Gateway IP address	The IP address of the gateway to the network for this route.

3. Select **Save**.

Assigning a static route to an existing node

To assign a static route to an existing Conferencing Node or Management Node:

1. Ensure that the static route that you want to apply has already been configured (**System > Static Routes**).
2. Go to **Platform > Conferencing Node** or **Platform > Management Node** as appropriate:
3. Select the node to which you want to assign the static route.
4. In the **Static routes** section, select from the list of **Available Static routes** the routes to assign to the node, and then use the right arrow to move the selected routes into the **Chosen Static routes** list.
5. Select **Save**.

Assigning a static route to a new Conferencing Node

Before you start the deployment process as described in [Deploying new Conferencing Nodes](#), ensure that the static routes that you want to apply have already been configured (**System > Static Routes**).

Then, while specifying the network configuration parameters to be applied to the new Conferencing Node, you will be able to include the static routes that you want to assign to that node.

Using event sinks to monitor conference and participant status

Information about the current status of any conferences that are in progress can normally be obtained using the [Management status API](#). However, in deployments with high levels of live system monitoring activity, such as those managed by service providers, frequent polling of the Management Node via the API can cause performance issues.

To avoid this you can configure the system to automatically send details of every participant and conference management event to an external service known as an **event sink**. When an event occurs, the Conferencing Node sends this information using a simple POST of JSON data to the nominated event sink server.


What information is sent?

To view which events are sent to an event sink, go to **System > Event Sinks** and from the bottom right of the page, select **View schema**. The same information can be downloaded as a Swagger or OpenAPI JSON document by selecting **Download Swagger schema** or **Download OpenAPI schema**.

You can also use a test site such as <https://webhook.site> to view live event data being sent.

Note that:

- Conference start and end event times are from the Conferencing Node's perspective and not the Management Node's perspective.
- You cannot control which events are sent to an event sink.

 For a complete description of the information that is sent, including examples, see [Event sink API](#).

Configuring event sinks

Each system location can be configured with one or more event sinks. The same event sink can be used for more than one system location.

To add, edit or delete an event sink, go to **System > Event Sinks**. The available options for each event sink are:

Option	Description
Name	The name used to refer to this event sink. Each event sink must have a unique name.
Description	An optional description of this event sink.
Version	Select which version of the API to use: <ul style="list-style-type: none">• <i>Version 1: Original</i>• <i>Version 2: Media streams</i> Version 2 contains additional <code>participant_media_stream_window</code> and <code>participant_media_streams_destroyed</code> participant events. Default: <i>Version 2</i>
URL	The URL of the external server to which events are sent.
Username and Password	The username and password used to authenticate to the external server when sending events. The username is case-sensitive. Leave these fields blank if authentication is not required.
Verify TLS	Whether to enable TLS verification when sending events. Only valid if the URL is HTTPS.
Location	The system locations in which to use this event sink.
Last attempted restart	This read-only field indicates if and when the event sink was last restarted.

Advanced configuration options

You can configure a range of advanced options that control connectivity and failure retry limits. These settings apply globally across all event sinks and are configured via **Platform > Global Settings > Advanced Event Sink Tuning**:

Option	Description
Event sink connection timeout	Maximum number of seconds allowed to connect to an event sink. Default: 7 seconds
Event sink maximum retry backoff	Maximum number of seconds allowed for the retry backoff before raising a "Eventsink Reached Maximum Backoff" alarm and stopping the event publisher. Default: 1800 seconds
Initial retry backoff	Initial time, in seconds, for the first retry attempt when an event cannot be delivered. Default: 1 second
Internal cache expiration	Internal cache expiration time in seconds. Used to briefly store "participant_disconnected" events in order to gather end-of-call media statistics. Default: 2 seconds
Time to wait for media streams message	Maximum time, in seconds, to wait for an end-of-call media stream message. Default: 1 second
Maximum number of background POSTs	Maximum number of incomplete background POSTs requests before stopping the event publisher and raising an "Eventsink Reached Maximum Concurrent POSTs" alarm. Default: 1000

Troubleshooting event sink failures

If an event cannot be delivered to an event sink, the node will try again after 1 second. If it fails again it tries again after 2 seconds, then 4, 8, 16 seconds and so on — it keeps doubling the timeout. In this case, the events may not necessarily be sent in sequence number (`seq` field) order.

If the timeout exceeds 30 minutes it will instead raise an "Eventsink Reached Maximum Backoff" alarm and stop the event sink publisher for that particular event sink.

Also, if more than 1000 events (configurable via **Maximum number of background POSTs**) are queued for an event sink but have not been sent then an "Eventsink Reached Maximum Concurrent POSTs" alarm is raised and the publisher for that event sink is stopped.

The retry/timeout parameters are configurable via **Platform > Global Settings > Advanced Event Sink Tuning** as described above.

To resolve event sink related alarms:

1. Check **support.event** logs for the reason for the event sink failures.
2. Take the appropriate action to resolve the failures.
3. Restart the event sink process:
 - a. Go to **System > Event Sinks** and select the event sink you want to restart.
 - b. Select **Restart**.

The event sink will restart within approximately 1-2 minutes (after the next Conferencing Node synchronization), and any backlogged events are sent.

Pexip Infinity platform configuration

Configuring the Management Node

The Management Node is the administrative interface of the Pexip Infinity platform, from which administrators can:

- Create and manage Conferencing Nodes.
- Configure Pexip Infinity services (Virtual Meeting Rooms, Virtual Receptions and so on).
- View platform and conference status across all Conferencing Nodes.
- Perform active conference management functions such as adding and disconnecting participants, enabling streaming or recording services, locking a conference, or muting a participant's audio.


The Management Node does not handle any conference media or signaling.


It is deployed using a virtual machine management application such as VMware's vCenter Server, or Microsoft Hyper-V, or on a cloud service such as Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP) or Oracle Cloud Infrastructure.

The initial configuration of the Management Node is provided via the installation wizard. Any subsequent changes to the configuration of the Management Node should be done using the Pexip Infinity Administrator interface or via the management API. Do not make any changes by any other means such as VMware or SSH; doing so may cause the Pexip Infinity service to fail. In particular you must not change the IP address of the Management Node.

To change configuration of the Management Node itself, go to **Platform > Management Node**.

The available options are:


Option	Description
Name	The name used to refer to the Management Node. It comprises the DNS Hostname and Domain suffix that were assigned to the Management Node during initial installation (when the installation wizard was run).
Description	An optional field to provide more information about the Management Node. This defaults to the Name but you can change it here.
DNS servers	Select one or more DNS servers to be used by the Management Node.  While you can assign unlimited DNS servers to a Management Node, only three will be used. They are used in the order in which they were assigned to the Management Node, with the first to be assigned having highest priority. If multiple servers are assigned simultaneously, those servers are used in descending numerical order. Hence, the order in which the DNS servers are prioritized is not necessarily the same as the order in which they are displayed.
NTP servers	Select one or more NTP servers to be used by the Management Node.
Web proxy	The web proxy to use for outbound web requests from this Management Node when sending usage statistics , incident reports , and some license activation requests, when communicating with the configured cloud bursting service, and (if One-Touch Join has been enabled and is using OAuth) when sending requests to the OAuth token endpoint.
TLS certificate	The TLS certificate to use on this node.
Configured FQDN	An optional identity for the Management Node. It is used by the web interface when indicating its own identity, for example when it needs to redirect to another page on itself. The name can be the same as its existing hostname.domain . If configured, the name must match an identity in the Management Node's TLS certificate.
IPv4 static NAT address	The IPv4 static NAT address for this Management Node. This allows IP-based access to a Management Node behind a NAT when it has an alternative Configured FQDN .
IPv6 address	The IPv6 address for the Management Node.

Option	Description
Gateway IPv6 address	<p>The IPv6 address of the default gateway.</p> <p>If this is left blank, the Management Node listens for IPv6 Router Advertisements to obtain a gateway address.</p>
Static routes	<p>From the list of Available Static routes, select the routes to assign to the node, and then use the right arrow to move the selected routes into the Chosen Static routes list.</p>
MTU	<p>(Maximum Transmission Unit) — the size of the largest packet that can be transmitted via the network interface of the Management Node. It depends on your network topology as to whether you may need to specify an MTU value here.</p> <p> If the Management Node is running in Google Cloud Platform, the MTU must not be higher than 1460 bytes.</p>
Enable SSH	<p>Determines whether this node can be accessed over SSH.</p> <p>Use Global SSH setting: SSH access to this node is determined by the global Enable SSH setting (Platform > Global Settings > Connectivity > Enable SSH).</p> <p>Off: this node cannot be accessed over SSH, regardless of the global Enable SSH setting.</p> <p>On: this node can be accessed over SSH, regardless of the global Enable SSH setting.</p> <p>Default: Use Global SSH setting.</p>

You can also change the Management Node's SNMP settings (see [Monitoring via SNMP](#) for more information):

Option	Description
SNMP mode	<p>Configures the SNMP access mode for the selected node:</p> <p>Off: SNMP is disabled. You cannot use SNMP to query the node for its status.</p> <p>SNMPv2c read-only: enables insecure, read-only access.</p> <p>SNMPv3 read-only: enables secure, read-only access, using the authPriv security level with SHA1 authentication and AES 128-bit encryption.</p> <p>When enabled, access is provided to the basic RFC 1213 MIB-II tree (1.3.6.1.2.1).</p> <p>Default: Off.</p>
SNMP community	<p>The SNMP group to which this node belongs. This setting applies to SNMPv2c only.</p> <p>Default: public</p>
SNMPv3 username	<p>The node's SNMPv3 username, used to authenticate SNMPv3 requests.</p>
SNMPv3 privacy password	<p>The node's SNMPv3 privacy password used for encrypting messages between the node and the management station.</p> <p>AES encryption must be used; DES is not supported.</p>
SNMPv3 authentication password	<p>The node's SNMPv3 authentication password, used to authenticate the associated username.</p> <p>The SHA authentication protocol must be used; MD5 is not supported.</p>
SNMP system contact	<p>The contact details (for example, email address) of the person responsible for this particular node.</p>
SNMP system location	<p>A description of the node's location.</p>

If you want SNMP traps to be sent from the Management Node to a particular SNMP Network Management System (NMS), select the NMS from the **SNMP NMS** drop-down menu:

Option	Description
SNMP NMS	The Network Management System to which the Management Node sends SNMP traps. If you have not already added the SNMP NMS , you can do so now by clicking  Pexip Infinity does not currently support traps with SNMPv3. If traps are required, use SNMPv2c.

Other details of the Management Node that cannot be changed via the Administrator interface are also shown on this page for your information, as follows:

Option	Description
IPv4 address	The IPv4 address of the Management Node.
Network mask	The IPv4 network mask of the Management Node.
Gateway IPv4 address	The IPv4 address of the default gateway.
Hostname	The DNS hostname of the Management Node.
Domain	The DNS domain of the Management Node.

If you need to change any of the hostname/addressing information, you must do so by [Re-running the installation wizard](#).


To perform other maintenance tasks such as changing the IP address of the Management Node or moving the Management Node to a different host server, see [Moving, restoring or changing the IP address of the Management Node](#).

To configure platform-wide settings, see [About global settings](#).

About global settings

Global settings are system-wide configuration options that affect the entire Pexip Infinity platform. Some of the settings may be overridden at the location, Conferencing Node or VMR level — this is indicated in the table below where applicable.

To configure the global settings, go to **Platform > Global Settings**. The settings are grouped into sections that you can **Show** or **Hide** individually, or by selecting **Expand all** or **Collapse all**, as follows:

 You should wait at least 90 seconds for any changes in configuration to be synchronized to all Conferencing Nodes; this may take longer in large deployments. You can go to **Status > Conferencing Nodes** to check when configuration was last updated.

Setting	Description	More information
Service configuration		
Guests-only timeout	The length of time (in seconds) for which a conference will continue with only Guest participants, after all Host participants have left. Default: 60 seconds	Using PINs to differentiate between Hosts and Guests
Last participant backstop timeout	The length of time (in seconds) for which a conference will continue with only one participant remaining. The type of participant (Host, Guest, automatically dialed, streaming etc) is irrelevant. The time can be configured to values between 60 seconds and 86400 (1 day), or to 0 (never eject). Default: 0 (never eject)	Automatically ending a conference

Setting	Description	More information
PIN entry timeout	The length of time (in seconds) for which a participant is allowed to remain at the PIN entry screen before being disconnected. Default: 120 seconds	Limiting the time a participant can spend at the PIN entry screen
Waiting for Host timeout	The length of time (in seconds) for which a Guest participant can remain at the waiting screen if a Host does not join, before being disconnected. Default: 900 seconds	Limiting how long Guests can wait for a Host
Default theme	The theme to use for services that have no specific theme selected.	Customizing conference images and voice prompts using themes
Maximum inbound call bandwidth (kbps)	Limits the bandwidth of media being received by Pexip Infinity from individual participants, for calls where bandwidth limits have not otherwise been specified.	Managing and restricting call bandwidth
Maximum outbound call bandwidth (kbps)	Limits the bandwidth of media being sent by Pexip Infinity to individual participants, for calls where bandwidth limits have not otherwise been specified.	Managing and restricting call bandwidth
Maximum call quality	Controls the maximum call quality for participants connecting to Pexip Infinity services. You can override this global setting for each individual service (VMR, Call Routing Rule etc). For example, you could use the default option of "HD" for most of your services by default, but enable Full HD on some specific services. The options are: <ul style="list-style-type: none"> SD: each participant is limited to SD quality. HD: each participant is limited to HD (720p) quality. Full HD (1080p): allows any endpoint capable of Full HD to send and receive its main video at 1080p. Default: HD	Setting and limiting call quality
Maximum presentation bandwidth ratio	When sending main video and presentation to a standards-based (SIP or H.323) endpoint, this defines the maximum percentage of the call bandwidth to allocate to the presentation content (with the remainder allocated to main video). It must be in the range 25% to 75%. Default: 75%	Managing and restricting call bandwidth
External participant avatar lookup	Determines whether or not avatars for external participants are retrieved using a method appropriate for the external meeting type. Currently this only applies to Microsoft Teams conferences. For all other conference types, and for when this option is not selected, avatars may be retrieved via external policy or user records as per standard behavior. You can also configure this setting on individual Call Routing Rules for Microsoft Teams conferences. Default: enabled.	
Connectivity		

Setting	Description	More information
Enable SIP *	Controls support for the SIP protocol over TCP and TLS across all Conferencing Nodes in your Pexip Infinity deployment. Note that disabling SIP will disable support for Skype for Business / Lync (MS-SIP). Default: enabled.	Enabling and disabling SIP, H.323, WebRTC and RTMP
Enable SIP UDP	Allows or prevents incoming calls over SIP UDP. Default: disabled.	
Enable H.323 * Enable WebRTC Enable RTMP *	These boxes control support for the selected protocols across all Conferencing Nodes in your Pexip Infinity deployment. Default: all of these settings are enabled by default.	
Enable support for Pexip Infinity Connect clients and Client API	Enables support for the Pexip Infinity client API. This is required for integration with the Infinity Connect browser-based, desktop and mobile clients, and any other third-party applications that use the client API, as well as for integration with Microsoft Teams. This setting must be enabled if you want to Enable WebRTC or Enable RTMP . Default: enabled.	
Enable Far End Camera Control	Allows endpoints that support FECC to be controlled by a Host participant using an Infinity Connect client. Default: enabled.	Using Infinity Connect in-call controls
Enable chat	Enables relay of chat messages between conference participants using Skype for Business / Lync and Infinity Connect clients. You can override this setting on a per conference basis (for a Virtual Meeting Room or Virtual Auditorium). Default: enabled.	Enabling and disabling chat messages
Enable outbound calls	Controls whether any calls can be made via the Infinity Gateway, and allows dial-out from a conference (via the Infinity Connect clients and the Administrator interface). Default: enabled.	Placing calls via the Pexip Infinity Distributed Gateway Manually dialing out to a participant from a conference
Enable legacy dialout API	This setting controls the system behavior when dialing out via an Infinity Connect client or the client API to a participant from an ongoing conference. When selected (enabled), calls placed via the: <ul style="list-style-type: none"> • Infinity Connect client always use automatic routing and thus must match an appropriate Call Routing Rule. • Client API or the legacy (webapp1) client can either use automatic routing or they can specify a dial-out protocol without any need for a Call Routing Rule i.e. it allows end-users to perform arbitrary dial outs (and thus circumvent any administrator-set rules). When not selected (disabled), calls that are placed via: <ul style="list-style-type: none"> • Any Infinity Connect client or via the client API always use automatic routing and thus must match an appropriate Call Routing Rule. Note that dial out via the Administrator interface, management API or Automatically Dialed Participants (ADPs) is unaffected by this setting. Default: disabled.	Manually dialing out to a participant from a conference

Setting	Description	More information
Do not default to the legacy Web App	When selected, Pexip Infinity does not default to the legacy version of the Infinity Connect web app. However, unsupported browsers will be redirected to the legacy web app. Default: enabled.	
Pexip Infinity domain (for Lync / Skype for Business integration)	The name of the SIP domain that is routed from Skype for Business / Lync to Pexip Infinity, either as a static route or via federation. You can also configure the Pexip Infinity domain on a per-location basis, which would override this global setting for Conferencing Nodes in that location.	Pexip Infinity and Microsoft Skype for Business / Lync Deployment Guide
Enable Skype for Business / Lync auto-escalation	When selected, this automatically escalates a Skype for Business / Lync audio call so that it receives video from a conference. Default: disabled.	Automatically escalating Skype for Business / Lync audio calls
Enable VbSS for Skype for Business	Controls support for Skype for Business Video-based Screen Sharing (VbSS). Note that VbSS is always enabled for Microsoft Teams calls, regardless of this setting. Default: disabled.	For information about enabling VbSS on your Skype for Business infrastructure see https://technet.microsoft.com/en-us/library/mt756736.aspx .
DSCP value for management traffic	The DSCP value for SSH, HTTPS and SNMP management traffic sent from the Management Node and from Conferencing Nodes. This is an optional Quality of Service (QoS) setting used to prioritize different types of traffic in large, complex networks. Also see Configuring system locations .	
Enable SSH	Allows an administrator to log in to the Management Node and all Conferencing Nodes over SSH. This setting can be overridden on individual nodes. Default: enabled.	
Enable directory	When disabled, Infinity Connect clients display aliases from their own call history only. When enabled, registered Infinity Connect clients additionally display the aliases of Virtual Meeting Rooms, Virtual Auditoriums, Virtual Receptions, and devices registered to the Pexip Infinity platform. Default: enabled.	Directory (phone book) of devices and VMRs for registered Infinity Connect clients
Enable restricted routing for Proxying Edge Nodes	When enabled, if a location only contains Proxying Edge Nodes, then those nodes only require IPsec connectivity with other nodes in that location, the transcoding location, the primary and secondary overflow locations, and with the Management Node. When disabled, a full connectivity mesh is required between all nodes in the deployment. Default: enabled.	Deployment guidelines for Proxying Edge Nodes

Setting	Description	More information
Media encryption	<p>Controls the media encryption requirements for participants connecting to Pexip Infinity services.</p> <p>You can override this global setting for each individual service (VMR, Call Routing Rule etc). For example, you could use the default option of "best effort" for most of your services, but enforce encryption on some specific services.</p> <ul style="list-style-type: none"> • Best effort: each participant will use media encryption if their device supports it, otherwise the connection will be unencrypted. • Required: all participants (including RTMP participants) must use media encryption. • No encryption: all H.323, SIP and MS-SIP participants must use unencrypted media. (RTMP participants will use encryption if their device supports it, otherwise the connection will be unencrypted.) <p>Default: Best effort</p>	
Port ranges		
Signaling port range start and end *	<p>The start and end values for the range of ports (UDP and TCP) that all Conferencing Nodes use to send signaling (for H.323, H.245 and SIP).</p> <p>Default: 33000–39999.</p>	
Media port range start and end *	<p>The start and end values for the range of ports (UDP and TCP) that all Conferencing Nodes use to send media for H.323, SIP, Skype for Business / Lync, WebRTC and RTMP (note that RTMP uses TCP only).</p> <p>Default: 40000–49999.</p>	
Codecs		
Codecs	<p>Controls which codecs to offer in audio/video negotiation (SDPs).</p> <p>Some third-party systems can experience issues if they are sent a large SDP from Pexip Infinity. You can reduce the size of the SDP by disabling specific, unwanted codecs.</p> <p>Default: all codecs are selected except AAC-LD128, H.264 High (mode 0) and H.264 High (mode 1).</p> <p>To enable the H.264 High Profile codec, move H.264 High (mode 1) into the list of Chosen Codecs. For optimal interoperability results, only enable H.264 High (mode 1) — leave H.264 High (mode 0) in the Available Codecs list.</p>	
Security		
OCSP state	<p>Determines whether OCSP is used to check the status of TLS certificates.</p> <p>Off: OCSP is not used.</p> <p>On: OCSP is used, and the request is sent to the URL specified in the TLS certificate. If no URL is specified in the TLS certificate, the OCSP responder URL configured below is used.</p> <p>Override: OCSP is used, and the request is sent to the OCSP responder URL specified in the OCSP responder URL field, regardless of any URL encoded in the TLS certificate.</p> <p>Default: Off.</p>	Using OCSP to check the status of certificates
OCSP responder URL	<p>The URL to which OCSP requests are sent if either:</p> <ul style="list-style-type: none"> • the OCSP state is set to On but no URL is present in the TLS certificate, or • the OCSP state is set to Override (in which case any URL present in the certificate is ignored). 	Using OCSP to check the status of certificates

Setting	Description	More information
SIP TLS certificate verification mode	<p>Determines whether to verify the peer certificate for connections over SIP TLS.</p> <p><i>Off</i>: the peer certificate is not verified; all connections are allowed.</p> <p><i>On</i>: the peer certificate is verified, and the peer's remote identities (according to RFC5922) are compared against the Application Unique String (AUS) identified by Pexip Infinity — the SIP URI — before the connection is allowed.</p> <p>Default: <i>Off</i>.</p>	Verifying SIP TLS connections with peer systems
Maximum log age (days)	<p>The maximum number of days of logs and call history to retain on Pexip nodes. On busy systems, logs may still be rotated before this time due to limited disk space.</p> <p>Enter 0 to have no set limit.</p> <p>Default: 0.</p>	
HTTP Content Security Policy	<p>Determines whether or not HTTP Content-Security-Policy (CSP) headers for Conferencing Nodes are enabled.</p> <p>Default: enabled.</p>	
HTTP Content Security Policy Header	<p>Defines the contents of the HTTP Content-Security-Policy headers for Conferencing Nodes when CSP is enabled.</p> <p>Default: upgrade-insecure-requests; default-src 'self'; frame-src 'self' https://telemetryservice.firstpartyapps.oaspapps.com/telemetryservice/telemetryproxy.html https://*.microsoft.com https://*.office.com; style-src 'self' 'unsafe-inline' https://*.microsoft.com https://*.office.com; object-src 'self'; font-src 'self' https://*.microsoft.com https://*.office.com; img-src 'self' https://www.adobe.com data: blob;; script-src 'self' 'unsafe-inline' 'unsafe-eval' https://*.microsoft.com https://*.office.com https://ajax.aspnetcdn.com https://api.keen.io; media-src 'self' blob;; connect-src 'self' https://*.microsoft.com https://*.office.com https://example.com;</p> <p>Note that these defaults are appropriate for typical usage of Pexip Infinity. For example, the Microsoft addresses are required for Outlook add-ins. The reserved domain example.com is included to support a third-party JavaScript library that is used by the Infinity Connect web app for PDF content sharing. You may need to add extra headers if you use custom plugins with the Infinity Connect web app.</p>	

Break-in resistance

Setting	Description	More information
Enable PIN brute force resistance	<p>Select this option to instruct Pexip Infinity to temporarily block all access to a VMR that receives a significant number of incorrect PIN entry attempts.</p> <p>You can override this setting on a per location basis.</p> <p>Default: enabled.</p>	Break-in resistance settings to mitigate rogue calls
Maximum PIN failures	<p>The maximum number of PIN failures allowed in any 10-minute window before the VMR is blocked.</p> <p>Default: 20.</p>	
Enable VOIP scanner resistance	<p>Select this option to instruct Pexip Infinity to temporarily block service access attempts from any unknown source IP addresses that dial a significant number of incorrect aliases.</p> <p>You can override this setting on a per location basis.</p> <p>Default: enabled.</p>	
Maximum scanner attempts	<p>The maximum number of incorrect dial attempts in any 10-minute window before the source IP address is blocked.</p> <p>Default 20.</p>	
External system integration		
Enable HTTP access for external systems	<p>Access for external systems is over HTTPS by default. If this box is selected, access over HTTP is also permitted.</p> <p>Default: disabled.</p>	Integrating with external systems
External system username and password	The username and password used by external systems (such as CUCM) when authenticating with Pexip Infinity.	Integrating with external systems
Management web interface configuration		
Login banner text	Any text entered here is displayed in a message box on the login page. This field supports plain text only.	
Enable management web interface session timeout	<p>Controls whether inactive users are automatically logged out of the Administrator interface after a period of time.</p> <p>If enabled, users are logged out after a number of minutes of inactivity as specified in the Management web interface session timeout setting.</p> <p>If disabled, users of the Administrator interface are never timed out. You may want to use this option if, for example, you have an administrator session that permanently monitors the system live view.</p> <p>Default: enabled.</p>	
Management web interface session timeout	<p>The number of minutes a browser session may remain idle before the user is logged out of the Management Node Administrator interface, if Enable management web interface session timeout is selected.</p> <p>Default: 30 minutes.</p>	
Reporting		

Setting	Description	More information
Enable incident reporting	If incident reporting is enabled, reports are sent to the specified URL.	Automatically reporting errors
Incident reporting URL	This setting is configured during initial installation of the Management Node (when running the installation wizard).	
Contact email address		
Automatically send deployment and usage statistics to Pexip	Select this option to allow submission of deployment and usage statistics to Pexip. This will help us improve the product. This setting is configured during initial installation of the Management Node (when running the installation wizard).	Automatically sending usage statistics
Advanced event sink tuning		
Event sink connection timeout	A range of advanced options to tune the event sink processes. See Using event sinks to monitor conference and participant status for details.	Using event sinks to monitor conference and participant status
Event sink maximum retry backoff		
Initial retry backoff		
Internal cache expiration		
Time to wait for media streams message		
Maximum number of background POSTs		
Cloud bursting		
Enable bursting to the cloud	These options enable and configure the Pexip Infinity platform for dynamic cloud bursting to either Microsoft Azure, Amazon Web Services (AWS) or Google Cloud Platform (GCP).	Dynamic bursting to the AWS cloud
Bursting threshold		Dynamic bursting to the Azure cloud
Tag name and value		Dynamic bursting to the Google Cloud Platform
Minimum lifetime		
Cloud provider		
Pexip Private Cloud		

Setting	Description	More information
Enable Pexip Private Cloud	Select this option to enable a connection from your deployment to the Pexip Private Cloud.	A connection to the Pexip Private Cloud is required if you wish to deploy a Pexip Smart Scale location .
Gateway URL	The URL used by your deployment to connect to the Pexip Private Cloud. This must be in the format <code>https://</code>	
Customer ID	The username used to authenticate your connection to the Pexip Private Cloud.	
Authentication token	The token used to authenticate your connection to the Pexip Private Cloud.	
Tech preview features		
Enable push notifications	This setting will support future feature updates to Infinity Connect mobile clients. If enabled, push notifications are sent to the external policy server whenever a registered mobile client receives a call, needs to refresh its session token, or becomes unregistered.	
Enable media relay on TCP port 443	This setting enables media relay on TCP port 443 on all Conferencing Nodes. This is intended as a fallback mechanism for use by WebRTC clients that are behind strict firewalls that block RTP media to Pexip's standard ports. This setting should only be enabled when it is impossible to amend the firewall's rules to allow UDP media, as sending media over TCP can result in increased latency and jitter. Enabling this setting may cause disruption to ongoing WebRTC sessions.	
Enable 1 + 33 layout	This setting makes the 1 + 33 layout available for use. This layout is suitable for large conferences. It displays 1 small main speaker and up to 33 other participants.	Selecting the layout seen by participants
* If you change any of these settings, all existing calls will be disconnected and all Conferencing Nodes will be automatically restarted.		

About system locations

System locations are typically used to group together Conferencing Nodes that are in the same physical location. System locations serve various purposes:

- They enable Pexip Infinity to make [intelligent decisions about routing](#) of media and signaling, including overflowing to another location when a particular location reaches its transcoding capacity, and the separation of [Proxying Edge Nodes](#) from Transcoding Conferencing Nodes.
- Services such as [DNS](#), [NTP](#), [H.323 gatekeepers](#), [SIP proxies](#), [web proxies](#), [Skype for Business / Lync Servers](#), Teams Connectors, [TURN servers](#), [STUN servers](#), [SNMP NMSs](#), [policy profiles](#), and [event sinks](#) are configured on a per-location basis, and are used by all Conferencing Nodes in that location.
- The [One-Touch Join](#) service also operates on a per-location basis.
- System locations can also be associated with [Pexip Smart Scale locations](#), which provide conferencing capacity in the Pexip Private Cloud.

A Conferencing Node's system location is assigned when the node is deployed, but it can be subsequently modified.

- i** If you change the system location of a Conferencing Node, all existing calls will be disconnected and the Conferencing Node will be restarted.

Intelligent and bandwidth efficient routing

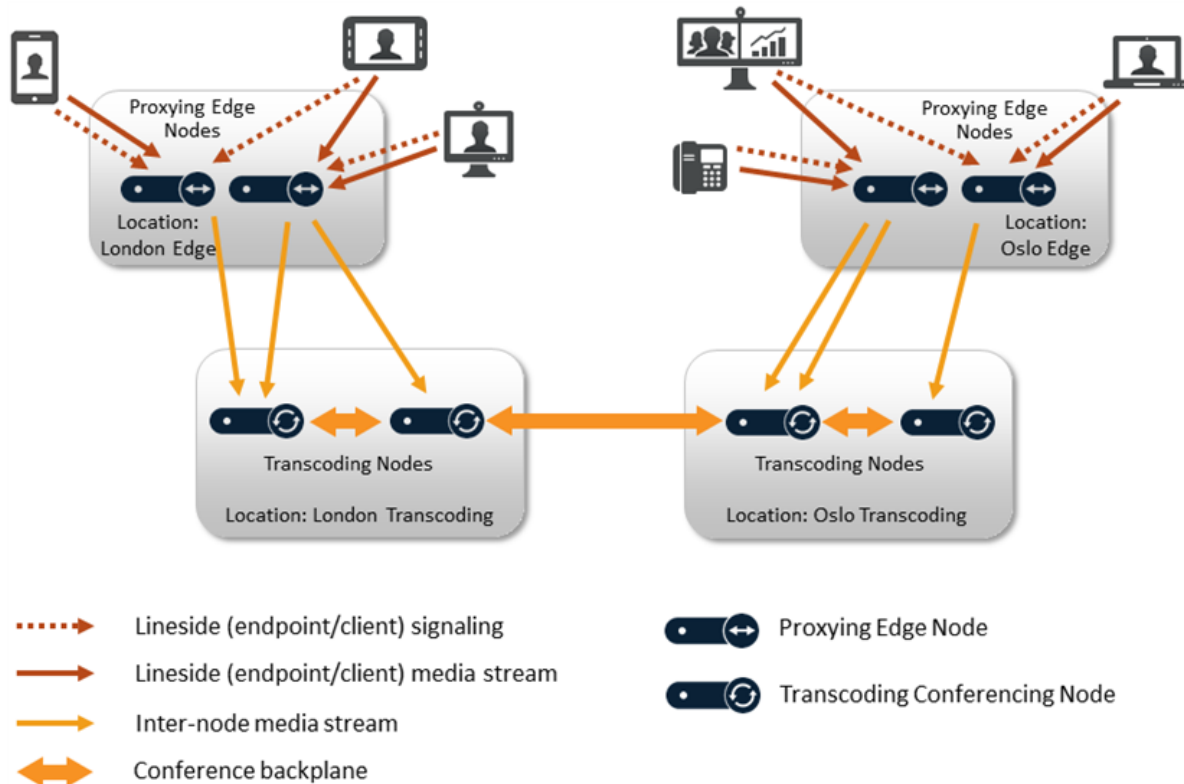
Grouping Conferencing Nodes by location allows Pexip Infinity to make intelligent decisions about routing. For example, if a conference is taking place across many Conferencing Nodes in two different locations, then Pexip Infinity will nominate one node in each location to act as an intermediary for that location. Media streams are sent between each intermediary only, rather than multiple streams being sent between each of the nodes at each of the locations. For more information, see [Conference distribution](#).

Calls that are received (for signaling purposes) can have their media received and transcoded by Conferencing Nodes within the same location as the node that is handling the signaling, or a different location can be nominated to handle the media. If the location handling the media reaches its transcoding capacity for a conference instance, additional "media overflow" locations can be utilized to handle the media for new conference participants that connect to nodes within the original signaling location. For more information, see [Media overflow locations](#).

Locations can also be configured to contain Proxying Edge Nodes that handle the signaling and the media connection with the calling device, but proxy the media onto another location for transcoding purposes (see [Deployments including Proxying Edge Nodes](#) below).

When grouping Conferencing Nodes into different locations, you should consider the amount of packet loss within your network. If there is a chance of packet loss due to network congestion between different groups of nodes, then they should be assigned separate system locations even if they are in the same physical location.

Deployments including Proxying Edge Nodes



You can deploy your Pexip Infinity platform as either a mix of Proxying Edge Nodes and Transcoding Conferencing Nodes, or as a system that only contains Transcoding Conferencing Nodes.

A typical deployment scenario is to use Proxying Edge Nodes as a front for many privately-addressed Transcoding Conferencing Nodes. Those outward-facing proxying nodes would receive all the signaling and media from endpoints and other external systems, and then forward that media onto other internally-located transcoding nodes to perform the standard Pexip Infinity transcoding, gatewaying and conferencing hosting functions.

A system location should not contain a mixture of proxying nodes and transcoding nodes. This separation of roles to locations simplifies load-balancing and conference distribution, and makes it easier for you to manage and monitor your externally-facing Proxying Edge Nodes distinctly from your Transcoding Conferencing Nodes. Hence, in the example scenario shown here, the Conferencing Nodes in the two locations "London Edge" and "Oslo Edge" are Proxying Edge Nodes and thus those nodes and locations are not involved in the actual hosting of any conferences. They forward the media onto the Transcoding Conferencing Nodes in the "London Transcoding" and "Oslo Transcoding" locations respectively.

Network deployment considerations

Nodes within a location actively synchronize with each other and may require a relatively high amount of network bandwidth for their communication. Pexip's backplane topology for distributed conferences assumes that there is a high availability, low latency, high-bandwidth network link between nodes within a location.

Therefore, while physical proximity is not a requirement, nodes in the same system location should typically be in the same physical datacenter or in physically proximate datacenters with an excellent link between them. There should be no more than 150 ms latency between Conferencing Nodes. If (as is often the case) you do not have such a network between your datacenters, we recommend that you consider assigning your nodes to different system locations.

In addition:

- Conferencing Nodes in a DMZ must not be configured with the same **System location** as nodes in a local network. This is to ensure that load balancing is not performed across nodes in the DMZ and nodes in the local network.
- Any Conferencing Nodes that are configured with a static NAT address must not be configured with the same **System location** as nodes that do not have static NAT enabled. This is to ensure that load balancing is not performed across nodes servicing external clients and nodes that can only service private IP addresses.
- We recommend that a **System location** should not contain a mixture of on-premises and cloud-hosted (Azure, AWS, GCP or Oracle) Conferencing Nodes.

See [Network deployment options](#) for more information about the various deployment scenarios.

Configuring services

Many platform and system services are configured on a per-location basis, and are used by all Conferencing Nodes in that location.

DNS and NTP servers

You must select at least one DNS server and at least one NTP server to be used by all of the Conferencing Nodes in that location. This allows you, for example, to assign nodes in a DMZ to a location that uses different DNS and NTP servers to a location containing nodes in a local, internal network.

H.323 gatekeepers, SIP proxies and Skype for Business / Lync servers

You can optionally specify the H.323 gatekeeper, SIP proxy, and Skype for Business / Lync server to use to route the outbound H.323/SIP/MSSIP call placed from a node within that location, when adding a new participant to a conference. These are the call control systems that are used when:

- a conference participant uses an [Infinity Connect client](#) to add another participant to the call
- the administrator [manually dials out to a participant from a Virtual Meeting Room](#)
- a participant is [automatically dialed out to from a Virtual Meeting Room](#)
- a third party uses the API to place a call to a participant

For more information, see [About H.323 gatekeepers and SIP proxies](#) and [About Skype for Business / Lync servers](#).

Web proxies

You can select a web proxy to use for some outbound web requests from all Conferencing Nodes in a location. When selected, the web proxy is used automatically for incident reporting, Epic telehealth requests, and for any One-Touch Join-related requests. For more information, see [Using a web proxy](#).

TURN servers and STUN servers

Pexip Conferencing Nodes can utilize a TURN server and negotiate TURN relays with the following ICE capable clients:

- Skype for Business / Lync clients
- WebRTC clients (the Infinity Connect web app on the latest browsers, and the desktop and mobile clients)

If these endpoints will be connecting to privately-addressed "on-premises" Conferencing Nodes, you **must** configure Pexip Infinity with the address of at least one TURN server that it can offer to ICE clients.

In some deployment scenarios where the TURN server is not located outside of the enterprise firewall, you may need to configure a separate STUN server so that each Conferencing Node can discover its public NAT address.

You can also configure the STUN servers to be used by Infinity Connect WebRTC clients when they connect to a Conferencing Node in this location.

For more information, see [Using TURN servers with Pexip Infinity](#) and [Using STUN servers with Pexip Infinity](#).

SNMP NMSs

If you have enabled SNMP support on one or more Conferencing Nodes in a particular location, you must also select the SNMP Network Management System (NMS) to which SNMP traps will be sent. The selected NMS is used for all Conferencing Nodes in the location that have SNMP support enabled.

Pexip Infinity does not currently support traps with SNMPv3. If traps are required, use SNMPv2c.

For more information, see [Monitoring via SNMP](#).

Policy profiles

Policy profiles specify how Pexip Infinity uses external policy and/or local policy to control its call policy and routing decisions. You can configure Pexip Infinity to use a different policy profile per system location.


For more information, see [Using external and local policy to control Pexip Infinity behavior](#).

Event sinks


In busy deployments where live event reporting is required, you can configure event sinks for each location. These are external service (s) to which Conferencing Nodes in this location send event information. For more information, see [Using event sinks to monitor conference and participant status](#).


Configuring system locations


To add, edit or delete system locations, go to **Platform > Locations**.

 You should wait at least 90 seconds for any changes in configuration to be synchronized to all Conferencing Nodes; this may take longer in large deployments. You can go to **Status > Conferencing Nodes** to check when configuration was last updated.

The available options are:

Option	Description
Name	The name you want to give to this physical location.
Description	An optional field where you can provide more information about the location.
DNS servers	<p>From the list of configured DNS servers, select one or more DNS servers to be used by all Conferencing Nodes in this location.</p> <p> While you can assign unlimited DNS servers to a location, only three will be used. They are used in the order in which they were assigned to the location, with the first to be assigned having highest priority. If multiple servers are assigned simultaneously, those servers are used in descending numerical order. Hence, the order in which the DNS servers are prioritized is not necessarily the same as the order in which they are displayed.</p>
NTP servers	From the list of configured NTP servers, select one or more NTP servers to be used by all the Conferencing Nodes in this location.
H.323 gatekeeper	The H.323 gatekeeper to use for outbound H.323 calls from this location, when adding an H.323 participant to a conference. For more information, see About H.323 gatekeepers and SIP proxies .
SNMP NMS	The Network Management System to which SNMP traps for all Conferencing Nodes in this location will be sent. For more information, see Monitoring via SNMP .
SIP proxy	The SIP proxy to use for outbound SIP calls from this location, when adding a SIP participant to a conference. For more information, see About H.323 gatekeepers and SIP proxies .

Web proxy	The web proxy to use for some outbound web requests from all Conferencing Nodes in this location. When selected, the web proxy is used automatically for incident reporting, Epic telehealth requests, and for any One-Touch Join-related requests. For more information, see Using a web proxy .
Lync / Skype for Business server	The Skype for Business / Lync server to use for outbound MS-SIP calls from this location, when adding a SfB/Lync participant to a conference. For more information, see About Skype for Business / Lync servers .
Microsoft Teams Connector	The Teams Connector to use for outbound calls to Teams meetings from this location, if a Virtual Reception or Call Routing Rule does not explicitly specify the Teams Connector to use.
TURN server	The TURN server to use when ICE clients (such as Skype for Business / Lync clients and Infinity Connect WebRTC clients) located outside the firewall connect to a Conferencing Node in this location. For more information, see Using TURN servers with Pexip Infinity .
STUN server	The STUN server to be used by all Conferencing Nodes in this location to determine the public IP address to signal to ICE clients (such as Skype for Business / Lync clients and Infinity Connect WebRTC clients) located outside the firewall. For more information, see Using STUN servers with Pexip Infinity .
Client STUN servers	The STUN servers to be used by Infinity Connect WebRTC clients when they connect to a Conferencing Node in this location. For more information, see Using STUN servers with Pexip Infinity .
MTU	<p>(Maximum Transmission Unit) — the size of the largest packet that can be transmitted via the network interfaces of the nodes in this location. It depends on your network topology as to whether you may need to specify an MTU value here.</p> <p> If any of the Conferencing Nodes in this location are running in Google Cloud Platform, the MTU must not be higher than 1460 bytes.</p> <p>Default: 1500</p>
DSCP value for media	<p>An optional setting used to prioritize different types of traffic in large, complex networks.</p> <p>This DSCP value tags the media traffic from Conferencing Nodes in this system location that is sent line side to endpoints and over the IPsec backplanes to other Pexip Conferencing Nodes.</p>
DSCP value for signaling	<p>An optional Quality of Service (QoS) setting used to prioritize different types of traffic in large, complex networks.</p> <p>This DSCP value tags the signaling traffic from Conferencing Nodes in this system location that is sent line side to endpoints and over the IPsec backplanes to other Pexip Conferencing Nodes.</p> <p>Note that some IPsec traffic between nodes — configuration synchronization and other non-realtime traffic — remains untagged.</p> <p>Also see DSCP value for management traffic in Global Settings.</p>

Transcoding location	<p>The system location to handle media transcoding for calls (signaling) received in, or sent from, this location.</p> <p>For calls received on a Proxying Edge Node, the media connection with the calling device is handled by a proxying node in this location, and the media is forwarded to a Transcoding Conferencing Node in the nominated Transcoding location (or an overflow location if necessary).</p> <p>For calls received on a Transcoding Conferencing Node, the media connection with the calling device and the transcoding is handled by a Transcoding Conferencing Node in the nominated Transcoding location.</p> <p>By default the transcoding location is set to <i>This location</i> i.e. the same location as where the call signaling is being handled, but you can change it to any of the other configured locations. You always need to choose a different transcoding location if this location contains Proxying Edge Nodes.</p> <p> All calls have to be transcoded somewhere. You should ensure that the selected location contains Transcoding Conferencing Nodes, otherwise calls will fail due to insufficient resources (unless you have also configured a primary or secondary overflow location).</p> <p>Note that if you change the media-handling locations for a proxying location (i.e. its transcoding, primary or secondary overflow locations), any proxied calls from that location that are currently being handled by the previously configured media locations will be dropped.</p> <p>See Handling of media and signaling for more information.</p>
Primary overflow location	An optional field where you can select the system location to handle media when capacity is reached in the Transcoding location , for calls (signaling) being handled in this location. For more information, see Media overflow locations .
Secondary overflow location	An optional field where you can select the system location to handle media when capacity is reached in both the Transcoding location and the Primary overflow location , for calls (signaling) being handled in this location.
Pexip Infinity domain (for Lync / Skype for Business integration)	<p>The name of the SIP domain that is routed from Microsoft Skype for Business / Lync to this Pexip Infinity location, either as a static route or via federation.</p> <p>This is an optional field. If configured, it is used instead of the global Pexip Infinity domain in outbound calls to Skype for Business / Lync from Conferencing Nodes in this location.</p>
Policy profile	The policy profile to be used by Conferencing Nodes in this location. For more information see Using external and local policy to control Pexip Infinity behavior .
Event sinks	The external service(s) to which Conferencing Nodes in this location send event information. For more information, see Using event sinks to monitor conference and participant status .
Enable PIN brute force resistance in this location	<p>Whether PIN brute force resistance is enabled for the Conferencing Nodes in this location:</p> <ul style="list-style-type: none"> • Use Global PIN brute force resistance setting: as per the global configuration setting. • No: PIN brute force resistance is disabled for nodes in this location. • Yes: PIN brute force resistance is enabled for nodes in this location. <p>When some locations have protection enabled, and other locations do not, the PIN brute force resistance setting is applied according to the location of the node that receives the call signaling.</p> <p>Default: <i>Use Global PIN brute force resistance setting</i>.</p>
Enable VOIP scanner resistance in this location	<p>Whether VOIP scanner resistance is enabled for the Conferencing Nodes in this location:</p> <ul style="list-style-type: none"> • Use Global VOIP scanner resistance setting: as per the global configuration setting. • No: VOIP scanner resistance is disabled for nodes in this location. • Yes: VOIP scanner resistance is enabled for nodes in this location. <p>When some locations have protection enabled, and other locations do not, the VOIP scanner resistance setting is applied according to the location of the node that receives the call signaling.</p> <p>Default: <i>Use Global VOIP scanner resistance setting</i>.</p>

Enabling Pexip Smart Scale

The Pexip Smart Scale (PSS) feature allows you to have Conferencing Nodes that are deployed by Pexip on your behalf within the secure Pexip Private Cloud, in the form of **Pexip Smart Scale locations**. You can add or remove these locations, and scale their capacity up or down, according to your own deployment's changing requirements.

With Pexip Smart Scale, you still deploy your own self-hosted Management Node and at least one Proxying Edge Node (which provides the call routing into your deployment). Then, instead of having to host all your own Transcoding Conferencing Nodes (which provide the processing capacity) as well, you can elect to have Pexip deploy some or all of them on your behalf, as Pexip Smart Scale locations. We create the necessary Conferencing Nodes (the actual number of which will depend on a variety of factors including the capacity required for that location), and these nodes then receive exactly the same service configuration from your Management Node as all the other Conferencing Nodes in your deployment.

Conferencing Nodes in a Pexip Smart Scale location can be seen as individual nodes within the **Status** pages of the Administrator interface. However, because these nodes are deployed on your behalf, you cannot edit any of their details. When you deploy a PSS location, you assign it to its own unique system location, and then configure that system location (e.g. DNS, NTP and other servers, web proxies, transcoding and overflow locations, policy profiles) in the usual way.

Available regions

Currently, Pexip Smart Scale runs in GCP; other cloud providers will be added in future releases. The regions in which you can deploy PSS locations are based on the cloud provider's regions; this is subject to change and we intend to add further regions over time. For information on what is currently available, please contact your Pexip authorized support representative.

Benefits

The benefits of using Pexip Smart Scale include:

- Privacy, security and control: you still own and control your Pexip Infinity deployment, network configuration, status and data (e.g. CDR); call signaling and media is terminated within your own deployment.
- Ease, flexibility and speed of deployment: nodes in Pexip Smart Scale locations are deployed remotely on your behalf by Pexip — we deal with the cloud service provider, calculate the resource requirements, and create and manage the required number of nodes in the locations of your choice. You can quickly increase or reallocate your resources depending on your changing requirements.

Use cases

New deployments

Pexip Smart Scale can be implemented within new Pexip Infinity deployments where you wish to retain complete control over your data and environment, but you don't have the necessary hardware or personnel to deploy and maintain your own conferencing capacity. In this case, you would deploy your own self-hosted Management Node and a number of self-hosted Proxying Edge Nodes (both of which do not require significant resources) and then use Pexip Smart Scale for your transcoding capacity (which requires relatively more powerful processing resource). This architecture minimizes the deployment overhead while ensuring that all calls (signaling and media) still terminate within the control of your own environment.

Expanding or migrating existing deployments

You can also implement Pexip Smart Scale within existing Pexip Infinity deployments where you wish to replace your existing hardware, or expand your capacity, without having to invest in your own additional hardware. This may be because your existing hardware is becoming outdated, you wish to migrate your physical datacenters to the cloud, you are seeing increases in your organization's use of videoconferencing, or you want to provide additional local capacity in specific regions.

In these cases, you can retain your existing Management Node and Proxying Edge Nodes, and then add Pexip Smart Scale locations, either in place of, or in addition to, your existing system locations.

Pexip Smart Scale versus cloud bursting

Pexip Smart Scale provides additional permanent capacity, provided and managed by Pexip, and paid for whether or not it is being used. [Cloud bursting](#) provides additional ad hoc capacity in a cloud service to which you must already have a subscription and for which you will be charged based on time used; you must manage the use of this capacity yourself.

Deployment summary

Deploying a Pexip Smart Scale location involves the following steps, described in the sections that follow:

- [Prerequisites](#)
- [Activating your connection to the Pexip Private Cloud](#)
- [Configuring the PSS system location](#)
- [Creating a Pexip Smart Scale location](#)
- [Configuring an associated Proxying Edge Node](#)
- [Configuring Call Routing Rules for PSS](#)
- [Configuring Automatically Dialed Participants for PSS](#)

Prerequisites

Network architecture and firewall considerations

Your Pexip Infinity deployment must include your own self-hosted Management Node and at least one self-hosted Proxying Edge Node. System locations used for Pexip Smart Scale must have connectivity to the Management Node and associated Proxying Edge Nodes, so you must configure appropriate bi-directional firewall rules. You may also need a VPN or similar connection between your LAN and the PSS location.

The diagram below gives a basic overview of a simple deployment, but the exact requirements will depend on your network and deployment architecture; please contact your Pexip authorized support representative for further information and guidance.

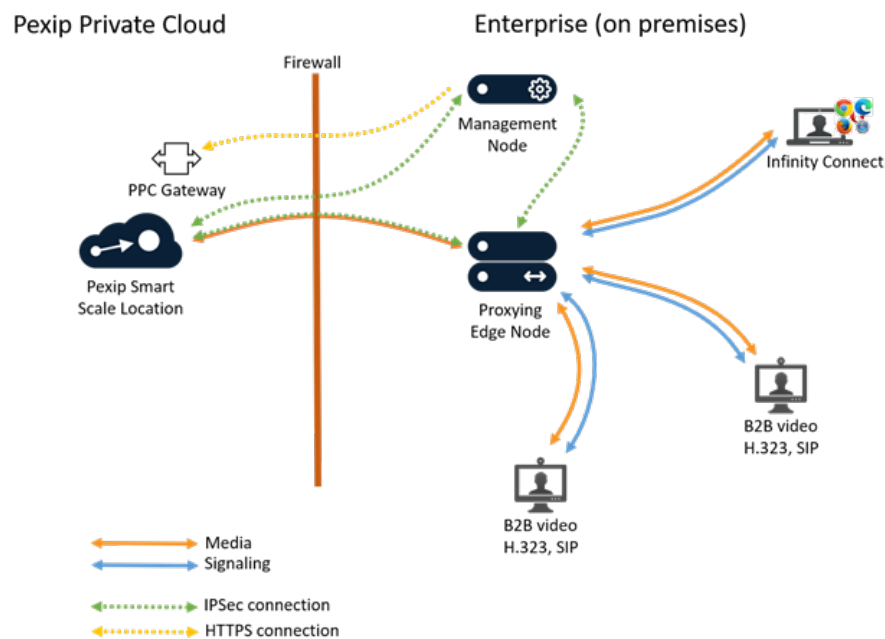


Diagram showing basic components of a Pexip Infinity deployment with Pexip Smart Scale

DNS and NTP access

Conferencing Nodes in a Pexip Smart Scale location do not have direct access to the internet, but will require access to a DNS server and an NTP server in order to function. You can provide access to these servers in one of the following ways, depending on the firewall policy within your deployment (in all cases, the required configuration is applied to the system location associated with the PSS location):

- use your own internal DNS and NTP servers, and allow traffic from the PSS nodes to these servers
- use the DNS and NTP servers provided by GCP: 169.254.169.254 (metadata.google.internal). In this case, no traffic from these servers will flow into or via your deployment.

Proxying Edge Node

Because PSS nodes are transcoding-only, you must have at least one Proxying Edge Node within your own self-hosted deployment to handle the signaling on behalf of PSS nodes. For more information, see [Configuring an associated Proxying Edge Node](#).

Media overflow locations and PSS

[Media overflow locations](#) are used to tell Pexip Infinity where to process the call media when there is no transcoding resource currently available in the [Transcoding location](#) associated with the location where the call signaling has been received. Because Pexip Smart Scale locations do not handle signaling, if you want to enable overflow when a PSS location has reached capacity, then you configure this on the system location that is handling the signaling on behalf of the PSS location (i.e. the system location containing the Proxying Edge Nodes that receive the calls). This system location should be configured to use the PSS system location as its [Transcoding location](#), and then use any other PSS locations (or other self-hosted locations containing transcoding nodes) in your deployment as its [Primary overflow location](#) and [Secondary overflow location](#), which will then be used if the PSS location reaches capacity.

Placement of outgoing calls from a Pexip Smart Scale location

Outgoing calls (calls placed from within a Virtual Meeting Room to another participant — such as those to an Automatically Dialed Participant, or when an administrator or Infinity Connect user adds another participant to the call) must be placed from a Conferencing Node within your own self-hosted deployment (rather than from a Pexip Smart Scale location). This is because Pexip Smart Scale locations are transcoding-only, and therefore do not handle call signaling.

To ensure that outgoing calls are not placed from a Pexip Smart Scale location, you must ensure your Call Routing Rules and Automatically Dialed Participants are configured as described in [Configuring Call Routing Rules for PSS](#) and [Configuring Automatically Dialed Participants for PSS](#) respectively.


Activating your connection to the Pexip Private Cloud

In this step, you activate your connection to the Pexip Private Cloud using the information provided to you by Pexip. To do this:

1. From the Pexip Infinity Administrator interface, go to **Platform > Global Settings**, and scroll down to the **Pexip Private Cloud** section.
2. Select **Enable Pexip Private Cloud**.
3. Enter the **Gateway URL**, **Customer ID** and **Authentication token** provided to you by Pexip.
4. Select **Save**.

Configuring the PSS system location

In this step you configure the the Pexip Infinity [system location](#) which will contain the Pexip Smart Scale location (which will itself be created in the next step).

 Each Pexip Smart Scale location must be contained with a single system location; that system location must not be used for any other PSS locations or contain any other Conferencing Nodes.

When configuring the system location that will contain the Pexip Smart Scale location, the available options are:

Option	Description
Name	The name you want to give to this Pexip Smart Scale system location. We recommend that you make it obvious in the name that this system location is used for Pexip Smart Scale.
Description	An optional field where you can provide more information about the location.

DNS servers	<p>From the list of configured DNS servers, select one or more DNS servers to be used by all Conferencing Nodes in this location. You may need to create specific firewall rules or use servers provided by GCP; see DNS and NTP access for more information.</p> <p>i While you can assign unlimited DNS servers to a location, only three will be used. They are used in the order in which they were assigned to the location, with the first to be assigned having highest priority. If multiple servers are assigned simultaneously, those servers are used in descending numerical order. Hence, the order in which the DNS servers are prioritized is not necessarily the same as the order in which they are displayed.</p>
NTP servers	<p>From the list of configured NTP servers, select one or more NTP servers to be used by all the Conferencing Nodes in this location. You may need to create specific firewall rules or use servers provided by GCP; see DNS and NTP access for more information.</p>
H.323 gatekeeper	Not required for Pexip Smart Scale system locations.
SNMP NMS	The Network Management System to which SNMP traps for all Conferencing Nodes in this location will be sent. For more information, see Monitoring via SNMP .
SIP proxy	Not required for Pexip Smart Scale system locations.
Web proxy	The web proxy to use for some outbound web requests from all Conferencing Nodes in this location. When selected, the web proxy is used automatically for incident reporting, Epic telehealth requests, and for any One-Touch Join-related requests. For more information, see Using a web proxy .
Lync / Skype for Business server	Not required for Pexip Smart Scale system locations.
Microsoft Teams Connector	Not required for Pexip Smart Scale system locations.
TURN server	Not required for Pexip Smart Scale system locations.
STUN server	Not required for Pexip Smart Scale system locations.
Client STUN servers	Not required for Pexip Smart Scale system locations.
MTU	<p>(Maximum Transmission Unit) — the size of the largest packet that can be transmitted via the network interfaces of the nodes in this location. For any system location being used for Pexip Smart Scale, the MTU must not be higher than 1460 bytes.</p> <p>Default: 1500</p>
DSCP value for media	<p>An optional setting used to prioritize different types of traffic in large, complex networks.</p> <p>This DSCP value tags the media traffic from Conferencing Nodes in this system location that is sent line side to endpoints and over the IPsec backplanes to other Pexip Conferencing Nodes.</p>
DSCP value for signaling	<p>An optional Quality of Service (QoS) setting used to prioritize different types of traffic in large, complex networks.</p> <p>This DSCP value tags the signaling traffic from Conferencing Nodes in this system location that is sent line side to endpoints and over the IPsec backplanes to other Pexip Conferencing Nodes.</p> <p>Note that some IPsec traffic between nodes — configuration synchronization and other non-realtime traffic — remains untagged.</p> <p>Also see DSCP value for management traffic in Global Settings.</p>
Transcoding location	<p>The system location to handle media transcoding for calls (signaling) received in, or sent from, this location.</p> <p>Not required for Pexip Smart Scale system locations. Leave as the default <i>This location</i>.</p>



Primary overflow location	Not required for Pexip Smart Scale system locations. See Media overflow locations and PSS for more information.
Secondary overflow location	Not required for Pexip Smart Scale system locations. See Media overflow locations and PSS for more information.
Pexip Infinity domain (for Lync / Skype for Business integration)	Not required for Pexip Smart Scale system locations.
Policy profile	Not required for Pexip Smart Scale system locations.
Event sinks	The external service(s) to which Conferencing Nodes in this location send event information. For more information, see Using event sinks to monitor conference and participant status .
Enable PIN brute force resistance in this location	Not required for Pexip Smart Scale system locations.
Enable VOIP scanner resistance in this location	Not required for Pexip Smart Scale system locations.

Creating a Pexip Smart Scale location

In this step you create the Pexip Smart Scale location, and configure it with the capacity you require, the geographical region in which you want it located, and the system location that will contain it.

Note that any changes to the configuration of a PSS location, including an increase or decrease of capacity, may result in the location being temporarily unavailable. We therefore suggest that any changes are made during maintenance periods.

1. From the Pexip Infinity Administrator interface, go to **Platform > Pexip Smart Scale**, and select **Add Pexip Smart Scale location**.
2. Configure the following:

Option	Description
Region	<p>From the drop-down list, select the geographical region in which you want to deploy the Pexip Smart Scale location.</p> <p>The options available in this list are based on the cloud provider's regions and are set up in advance by Pexip. If you want to change the available options, please contact your Pexip authorized support representative.</p>
Max HD calls	<p>This setting determines the capacity of the Pexip Smart Scale location you are creating. Capacity is defined in terms of HD calls, and therefore this location will support a higher number of SD and audio calls, or a lower number of Full HD calls.</p> <p> You can increase the capacity of a region at any time. However, you must wait 24 hours after any configuration change to decrease the capacity.</p>
System location	<p>This is the Pexip Infinity system location (created in the previous step) in which this Pexip Smart Scale location will exist. A system location used for PSS should not include any other Conferencing Nodes.</p> <p>You can create and configure a new system location by selecting the plus  symbol.</p>

3. Select **Save**.
4. You must now enable the location. To do this, go back to **Platform > Pexip Smart Scale**, select the PSS location you have just created, and from the **Action** drop-down list, select **Enable configuration** and then select **Go**.

It may take up to 20 minutes for an initial PSS location to become available. When it has done so successfully, its **Status** will change to *Deployed*.

Configuring an associated Proxying Edge Node

Pexip Smart Scale locations are transcoding-only; they do not handle any signaling. Therefore every PSS location must have at least one node — usually a Proxying Edge Node — that will handle signaling on its behalf.

These Proxying Edge Nodes:

- will receive and place calls, handle signaling and terminate media on behalf of the Pexip Smart Scale location
- must reside within your own self-hosted deployment (either on-premises, or within your own cloud subscription)
- must be placed in a separate system location to that of the PSS location.

The system location containing these Proxying Edge Nodes:

- must be configured to use the PSS location as its **Transcoding location**
- could also optionally be configured with a primary or secondary overflow location, in case you run out of capacity in the PSS location. These overflow locations could be other PSS locations in a different region, or a location containing your own self-hosted Conferencing Nodes. For more information, see [Media overflow locations and PSS](#).

If you have not already done so, you should now:

1. Create at least one Proxying Edge Node.
2. Create a system location that contains the Proxying Edge Node.
3. Configure that system location's **Transcoding location** to use the system location used for Pexip Smart Scale.

For more information, see [Deployment guidelines for Proxying Edge Nodes](#) and [About system locations](#).

Configuring Call Routing Rules for PSS

In Pexip Infinity deployments that include Pexip Smart Scale locations, you must ensure that outgoing calls are not placed from a Pexip Smart Scale location. To do this:

- you should ensure that none of your [Call Routing Rules](#) have a PSS location as their designated **Outgoing location**
- you should configure your Automatically dialed participants [as described below](#)
- when [dialing out to participants manually](#) (via the Administrator interface or management API), ensure that calls are not placed from a PSS location.

To make doubly sure that there are no attempts to place outgoing calls from a Pexip Smart Scale location, you may wish to create a [Call Routing Rule](#) that routes all outgoing calls from a Pexip Smart Scale location via its Proxying Edge Node location. Such a rule should include the following settings:

Option	Input	Notes
Use this rule for...		
Incoming gateway calls	Do not select this option	
Outgoing calls from a conference	Select this option	
Calls being handled in location	Select the Pexip Smart Scale system location	
Alias match and transform		
Match against full alias URI	Do not select this option	
Destination alias regex match	.+@.\+\.+	This regular expression will match any destination alias at any domain.
Destination alias regex replace string	<leave blank>	We have left this blank because we do not want to amend the alias.
Outgoing call placement		
Outgoing location	We recommend that you select the system location of the Proxying Edge Node (s) that handle the signaling for the Pexip Smart Scale system location.	

Configuring Automatically Dialed Participants for PSS

To ensure that outgoing calls are not placed from a Pexip Smart Scale location, all [Automatically Dialed Participants](#) in your deployment (regardless of whether the ADP has *Route this call* set to *Manually* or *Automatically*) must have their **Outgoing location** configured to use a specific system location within your own self-hosted deployment that contains either Proxying Edge Nodes or Transcoding Conferencing Nodes. In other words, the **Outgoing location** for ADPs **must not** be set to *Automatic*, or to any PSS location.

In most cases, we recommend that the **Outgoing location** is set to the Proxying Edge Node location that is handling the signaling for the PSS location. Alternatively (for example, for larger or geographically distributed deployments) you could select a Proxying Edge Node or a Transcoding Conferencing Node that is geographically close to the location of the endpoint to be dialed.

If the ADP has *Route this call* set to *Automatically*, you must also ensure that all calls placed from the selected **Outgoing location** match a Call Routing Rule — this can either be a general rule, or a rule specific to that location.

About H.323 gatekeepers and SIP proxies

You can configure the Pexip Infinity platform with the addresses of one or more H.323 gatekeepers and SIP proxies. These are the call control systems that can be used to route outbound H.323 and SIP calls on behalf of Pexip Infinity.

i The H.323 gatekeepers and SIP proxies configured here are used for **outbound calls from** Pexip Infinity. They do not determine the systems used to route **inbound calls to** Pexip Infinity. Pexip Infinity **will not** automatically register with any systems configured here. To route inbound calls from a gatekeeper or SIP proxy, you must configure these systems with neighbor zones that direct calls to Pexip Infinity. For more information see [Call control](#).

Outbound calls are made from Pexip Infinity when:

- a conference participant uses an [Infinity Connect client](#) to add another participant to the call
- the administrator [manually dials out to a participant from a Virtual Meeting Room](#)

- a participant is [automatically dialed out to from a Virtual Meeting Room](#)
- a third party uses the API to place a call to a participant
- the [Infinity Gateway](#) is used to interwork a call.

Nominating which H.323 gatekeeper and SIP proxy to use

To configure H.323 gatekeepers, go to **Call Control > H.323 Gatekeepers**, and to configure SIP proxies, go to **Call Control > SIP Proxies**.

When configuring the proxy or gatekeeper, the target **Address** can be an IP address or an FQDN, and for a SIP proxy you must also specify the transport protocol.

If the target address is an FQDN, Pexip Infinity looks for DNS SRV records first if the **Port** field is blank. If a port is specified then Pexip Infinity only performs a DNS A / AAAA lookup. When using SRV records, if multiple records are returned they are used in order based on priority and weight, and if the connection to the target host(s) fails then the host(s) associated with the target of the next SRV record are tried (as per RFC3263 for SIP and H.323 Annex O).

Example DNS SRV records for locating H.323 gatekeepers and SIP proxies

This example shows the DNS SRV records required for H.323 gatekeepers and SIP proxies, which in this case are two Cisco VCSs in the vcs.example.com domain. In this example there are two h323ls service records for the two target VCS gatekeepers, and there is a set of two service records for the two target VCS SIP proxies for each transport protocol (TCP, TLS i.e. sips and UDP). Note that if the SIP proxy or gatekeeper is hosted in a remote environment, you (as the administrator for your own domain) will not have any authority over the DNS records for that remote domain — and are not responsible for creating them — you should check that such records exist when troubleshooting any outbound calling issues.

Name	Service	Protocol	Priority	Weight	Port	Target host
vcs.example.com.	h323ls	udp	10	10	1719	vcs01.vcs.example.com.
vcs.example.com.	h323ls	udp	10	10	1719	vcs02.vcs.example.com.
vcs.example.com.	sip	tcp	10	10	5060	vcs01.vcs.example.com.
vcs.example.com.	sip	tcp	10	10	5060	vcs02.vcs.example.com.
vcs.example.com.	sips	tcp	10	10	5061	vcs01.vcs.example.com.
vcs.example.com.	sips	tcp	10	10	5061	vcs02.vcs.example.com.
vcs.example.com.	sip	udp	10	10	5060	vcs01.vcs.example.com.
vcs.example.com.	sip	udp	10	10	5060	vcs02.vcs.example.com.

In your actual deployment, both the **Name** and the **Target host** should be changed to use the real domain name and host names of the call control systems.

i After adding the details of the H.323 gatekeeper or SIP proxy, you must then associate it with the relevant location (**Platform > Locations**) or Call Routing Rule (**Services > Call Routing**).

- Each [System location](#) can have a nominated H.323 gatekeeper and SIP proxy — these are used when adding a new H.323 or SIP participant to a conference and define where to route the outbound H.323/SIP calls placed from nodes within that location.
- Each [Call Routing Rule](#) can have a nominated H.323 gatekeeper or SIP proxy — these are used to route the outbound leg of gateway calls for rules that are targeted at external H.323 and SIP systems.

If an Infinity Connect user adds an H.323 or SIP participant to a conference, the call is routed to the H.323 gatekeeper or SIP proxy that is associated with the system location of the Conferencing Node to which the Infinity Connect client is connected. If the location does not have a configured H.323 gatekeeper or SIP proxy, Infinity Connect users connected to Conferencing Nodes in that location may still be able to dial in other participants to a conference — to enable this, the user must enter the participant's IP address or FQDN (and for the latter, DNS must be configured properly), thus allowing Pexip Infinity to dial the participant directly.

About Skype for Business / Lync servers

Pexip Infinity uses Skype for Business / Lync* servers to route **outbound** MS-SIP calls.

Outbound calls are made from Pexip Infinity when:

- a conference participant uses an [Infinity Connect client](#) to add another participant to the call
- the administrator [manually dials out to a participant from a Virtual Meeting Room](#)
- a participant is [automatically dialed out to from a Virtual Meeting Room](#)
- a third party uses the API to place a call to a participant
- the [Infinity Gateway](#) is used to interwork a call.

You can configure the addresses of one or more SfB/Lync servers within Pexip Infinity. These can be Front End Servers or a Director; they cannot be Edge Servers.

Within an **on-prem** Skype for Business / Lync environment, you must then associate a specific SfB/Lync server with a [System location](#), [Call Routing Rule](#) or [Virtual Reception](#) to instruct Pexip Infinity to use that SfB/Lync server when it places an outbound MS-SIP call from a Conferencing Node in that location, or when it matches that routing rule.

Within a **public DMZ** Skype for Business / Lync deployment, you must ensure that a Pexip Infinity location is **not** configured to route calls to a specific SfB/Lync server. This is to ensure that each Conferencing Node uses DNS to locate an appropriate SfB/Lync system via which to route outbound calls.

To add, edit or delete SfB/Lync servers, go to **Call Control > Lync / Skype For Business Servers**.

The available options are:

Option	Description
Name	The name used to refer to this SfB/Lync server in the Pexip Infinity Administrator interface.
Description	An optional description of the SfB/Lync server.
Address	The IP address or FQDN of the SfB/Lync server. This can be a Front End Server or Director; it must not be an Edge Server.
Port	The IP port on the SfB/Lync server to which the Conferencing Node will connect. Default: 5061.
Transport	The IP transport used to connect to the SfB/Lync server. Default: <i>TLS</i> .

To associate a SfB/Lync server with a System location, go to **Platform > Locations**, to associate it with a Call Routing Rule, go to **Services > Call Routing**, and to associate it with a Virtual Reception that is acting as an IVR gateway to scheduled and ad hoc SfB/Lync meetings, go to **Services > Virtual Receptions**.

For more information on integrating Pexip Infinity with SfB/Lync, see [Using Microsoft Skype for Business / Lync with Pexip Infinity](#).

* Note that where this documentation refers to "SfB/Lync", it represents both Microsoft Skype for Business and Lync unless stated otherwise.

Using TURN servers with Pexip Infinity

A TURN server is a media relay/proxy that allows peers to exchange UDP or TCP media traffic whenever one or both parties are behind NAT.

ICE (Interactive Connectivity Establishment) is a framework that allows endpoints to discover paths through which they can exchange media — such as via a TURN server — where direct routing between those devices may not be possible, e.g. when a device is on a private network or is behind a NAT.

Pexip Conferencing Nodes can utilize a TURN server and negotiate TURN relays with the following ICE capable clients:

- Skype for Business / Lync clients
- WebRTC clients (the Infinity Connect web app on the latest browsers, and the desktop and mobile clients)

If these endpoints will be connecting to privately-addressed "on-premises" Conferencing Nodes, you **must** configure Pexip Infinity with the address of at least one TURN server that it can offer to ICE clients.

A TURN server is not required if your Conferencing Nodes are publicly-addressable, either directly or via static NAT. For more information, see [When is a reverse proxy, TURN server or STUN server required?](#).

Note that the H.323 specification has no concept of ICE.

The TURN servers that you use with Pexip Infinity:

- Must have a public address (located either on the public internet or in a DMZ).
- Unless a separate [STUN server](#) has also been configured, they must be deployed in such a way that traffic from the Conferencing Node towards the TURN server appears as coming from the Conferencing Node's public NAT address (its server reflexive address), otherwise this will prohibit some Skype for Business / Lync communication scenarios.
- Must be routable from your Conferencing Nodes.
- Must be standards-based (supporting [RFC 5766](#)), for example a Pexip TURN server (see [Pexip Reverse Proxy and TURN Server Deployment Guide](#)) or a VCS Expressway.

When using a TURN server with a Conferencing Node:

- Conferencing Nodes only use TURN over UDP (not TCP). However, Conferencing Nodes will perform ICE TCP negotiation.
- Conferencing Nodes always communicate with its configured TURN server over a single UDP port (default UDP/3478). UDP media is multiplexed from the Conferencing Node to that single port on the TURN server. The TURN server will reply back to the same port pair on the Conferencing Node. The TURN server never initiates a connection towards a Conferencing Node.
- As general good practice, we always recommend deploying the TURN server in a suitably secured network segment, such as a DMZ.

Note that Microsoft A/V Edge Server does not support RFC 5766 and therefore cannot be used as a TURN server with Pexip Infinity.


Nominating the TURN servers used by Pexip Infinity

Within Pexip Infinity you specify one or more TURN servers, and then associate one of those TURN servers with each [System location](#) and [Call Routing Rule](#). The same TURN server can be used by more than one location or rule.

- To add, edit or delete TURN server connection details, go to **Call Control > TURN Servers**. The options are:

Option	Description
Name	The name used to refer to this TURN server in the Pexip Infinity Administrator interface.
Description	An optional description of the TURN server.
Address	The IP address or FQDN of the TURN server.
Port	<p>The IP port on the TURN server to which the Conferencing Node will connect.</p> <p>Default: 3478.</p> <p>When a port is specified, Pexip Infinity performs a DNS A-record lookup on the Address.</p> <p>If a port is not specified, then a _turn._udp DNS SRV lookup on the Address is performed (and the port in the SRV record is used). There is no fallback to an A record lookup if the SRV lookup fails.</p>
Username and Password	The credentials of an account on the TURN server that can be used to create bindings.

- To associate a TURN server with a location, go to **Platform > Locations** and configure the relevant location.
- To associate a TURN server with a Call Routing Rule, go to **Services > Call Routing** and configure the relevant rule.

 After adding the details of the TURN server, you must add it to the relevant locations and Call Routing Rules.

How Pexip Infinity decides which TURN server to offer

The TURN server offered by Pexip Infinity to ICE clients is determined as follows:

- **Conferences:** uses the TURN server associated with the location of the Conferencing Node that is handling the call signaling.
- **Point-to-point calls via the Infinity Gateway:** uses the TURN server associated with the Call Routing Rule that matched the call request. If there is no TURN server associated with the rule, then the TURN server associated with the location of the Conferencing Node that is handling the call is used instead. Note that rules can optionally be configured on a per-location basis.

If a TURN server is not configured for the location or rule, a TURN server relay candidate will not be offered.

When a Conferencing Node receives a call from an ICE client, it sends a request to the TURN server to allocate bindings to be used by that client. The client uses these bindings to route its call media through the firewall to the Conferencing Node. The Conferencing Node allocates up to four TURN bindings per call (made up of two TURN bindings per media stream for both audio and video).

Using STUN servers with Pexip Infinity

A STUN server allows clients, such as Conferencing Nodes or Infinity Connect WebRTC clients, to find out their public NAT address.

When a client is deployed behind a NAT, it can send a STUN request to the STUN server, which responds back to the client and tells it from which IP address it received the STUN request. Using this method, the client can discover its public NAT address, which is important in order for ICE to work between Conferencing Nodes and other ICE-enabled clients (for example, WebRTC and Skype for Business / Lync clients). In relation to ICE, this public NAT address is also known as the server reflexive address.

In Microsoft Skype for Business and Lync deployments it is essential that a Conferencing Node can discover its public NAT address.

Conferencing Nodes

If a Conferencing Node is deployed on a private network behind a NAT, its system location may already be configured with the details of a [TURN server](#) (such as the Pexip TURN server). Often, that TURN server can act as a STUN server and a separate STUN server is not normally required.

By default, Conferencing Nodes send their STUN requests to the TURN server, but if the TURN server is not located outside of the enterprise firewall then the Conferencing Node will not be able to discover its public NAT address. If this is the case in your deployment scenario, you must configure a separate STUN server — the Conferencing Node's STUN requests will then be sent to the STUN server, instead of the TURN server.

A STUN server is not required if:

- your Conferencing Nodes are publicly-addressable, either directly or via static NAT, or
- the STUN requests sent from the Conferencing Nodes to the TURN server will return the public NAT address of the Conferencing Node.

The STUN servers used by Pexip Infinity must be located outside of the enterprise firewall and must be routable from your Conferencing Nodes.

Infinity Connect WebRTC clients

When connecting to a privately-addressed Conferencing Node, Infinity Connect WebRTC clients that are behind a NAT may also use a STUN server to find out their public NAT address.

When an Infinity Connect WebRTC client connects to a Conferencing Node, the node will provision it with any **Client STUN server** addresses that are configured against that node's system location. The WebRTC client can then use those STUN servers to discover its public NAT address. This is typically only required if the WebRTC client is communicating via a TURN server.

For more information, see [When is a reverse proxy, TURN server or STUN server required?](#).

How Conferencing Nodes decide which STUN server to use

The STUN server used by a Pexip Infinity Conferencing Node handling a call is determined as follows:

- **Conferences:** uses the STUN server associated with the location of the Conferencing Node that is handling the call signaling.
- **Point-to-point calls via the Infinity Gateway:** uses the STUN server associated with the Call Routing Rule that matched the call request. If there is no STUN server associated with the rule, then the STUN server associated with the location of the Conferencing Node that is handling the call signaling is used instead. Note that rules can optionally be configured on a per-location basis.

If a STUN server is not configured for a location or rule, but a TURN server is configured, the Conferencing Node will send STUN requests to that TURN server.

Nominating the STUN servers used by Pexip Infinity and Infinity Connect WebRTC clients

Within Pexip Infinity you can configure the addresses of one or more STUN servers. You then associate those STUN servers with each [System location](#) (with separate configuration for the STUN server used by Conferencing Nodes in that location, and the STUN servers to offer to Infinity Connect clients connected to that Conferencing Node), and with each [Call Routing Rule](#).

Configuring STUN server addresses

To add, edit or delete STUN server connection details, go to **Call Control > STUN Servers**. The options are:

Option	Description
Name	The name used to refer to this STUN server in the Pexip Infinity Administrator interface.
Description	An optional description of the STUN server.
Address	The IP address or FQDN of the STUN server. This should not be the same address as any of your configured TURN servers.
Port	The IP port on the STUN server to which the Conferencing Node will connect. Default: 3478.

Note that Pexip Infinity ships with one STUN server address already configured by default: **stun.l.google.com**. This STUN server uses port 19302 (rather than the common 3478) and can be assigned to system locations for use by Infinity Connect WebRTC clients.

Associating STUN server addresses with Conferencing Nodes

To associate a STUN server address with a Conferencing Node, you must configure the node's system location:

1. Go to **Platform > Locations**.
2. Select the Conferencing Node's location.
3. Select a **STUN server** and select **Save**.

All Conferencing Nodes in that location will use the nominated STUN server for conference calls.

Associating STUN server addresses with gateway calls

If a gateway call is being placed to an ICE-enabled client (such as Skype for Business / Lync clients and Infinity Connect WebRTC clients), the Conferencing Node placing the call will use the STUN server associated with the matching rule. (For gateway calls, the Conferencing Node does not use the STUN server associated with its system location.)

To associate a STUN server address with a Call Routing Rule:

1. Go to **Services > Call Routing**.
2. Select the relevant rule.
3. Select a **STUN server** and select **Save**.

Configuring the STUN server addresses provided to Infinity Connect WebRTC clients

To configure the specific STUN server addresses that are provisioned to Infinity Connect WebRTC clients, you must configure the system locations used by the Conferencing Nodes that the clients connect to:

1. Go to **Platform > Locations**.
2. Select the Conferencing Node's location.
3. Select one or more **Client STUN servers** and select **Save**.

When an Infinity Connect WebRTC client connects to a Conferencing Node in that location, the Conferencing Node will provide it with the addresses of the nominated STUN servers. These STUN servers are used by the client to discover its public NAT address.

If no **Client STUN servers** are configured for that node/location, the Infinity Connect client may still be able to communicate by using a TURN relay, if a TURN server is configured on the Conferencing Node, but this may cause delays in setting up media.

For clients on the same network as the Conferencing Nodes, where no NAT is present, users may find that WebRTC call setup time is improved by removing all **Client STUN servers**.

Configuring policy profiles

Policy profiles specify how Pexip Infinity uses external policy and/or local policy to control its call policy and routing decisions.

Each policy profile can be used to:

- control which types of data (e.g. service configuration, media location, participants avatars etc.) are managed via policy — either by external policy, or by local policy (where local policy is supported for that data type) or by both external and local policy
- nominate the address of an external policy server to which the external policy API requests are sent
- specify the local policy jinja2 script to be executed against the data (currently service configuration and media location data types only).

You can configure Pexip Infinity to use both external and local policy depending on your requirements. When both external and local policy are enabled, external policy is applied first to retrieve the configuration data from the external system, and then local policy is applied to that retrieved data (which can then conditionally modify that data). See [Using external and local policy to control Pexip Infinity behavior](#) for more information.

Each system location is configured with a policy profile and that profile is then used by all of the Conferencing Nodes in that location whenever they need to retrieve configuration data. This means that you could use the same policy profile in all locations (and thus all Conferencing Nodes), or if required you can configure many different profiles with, for example, different local policy scripts or different external policy server URIs, and then assign different policy profiles to different system locations.

i You must assign policy profiles to locations otherwise they will never be used. If you want to configure just one policy profile to be used globally you need to assign it to all of your locations.

When using external policy within a system location, you must ensure that each Conferencing Node in that location is able to reach the nominated policy server.

To configure policy profiles:

1. Go to **Call Control > Policy Profiles**.
2. Select **Add Policy profile** and then configure that profile. The options are:

Option	Description
Name	The name used to refer to this policy profile in the Pexip Infinity Administrator interface.
Description	An optional description of the policy profile.
External policy server	
URL	<p>The URL of the policy server to use for all external policy API requests from this profile, for example https://policy.example.com/path.</p> <p>You can only configure one address URL per policy server.</p> <p>If the request is over HTTPS, Pexip Infinity must trust the certificate presented by the policy server.</p>
Username	Optional fields where you can specify the credentials required to access the external policy server.
Password	External policy requests support Basic Authentication and basic ASCII-encoded usernames and passwords.
Avatar policy	
Use local avatar configuration	When Use local avatar configuration is enabled, requests to fetch avatar images to represent directory contacts and conference participants are sent to the Avatar URL associated with the user configured within Pexip Infinity.
Enable external avatar lookup	<p>If enabled, requests are sent to the external policy server to fetch avatar images to represent directory contacts and conference participants.</p> <p>If both Use local avatar configuration and Enable external avatar lookup are enabled, then the local avatar configuration takes precedence. However, if no matching user record is found, or the user record does not have a configured Avatar URL then a request is made to the external policy server instead. If there is an Avatar URL, and the request fails for any reason, Pexip Infinity will not fall back to external policy.</p>
Service configuration policy	

Option	Description
Enable external service configuration lookup	If enabled, requests are sent to the external policy server to fetch service configuration data (VMRs, Virtual Receptions, Infinity Gateway calls etc).
Apply local policy	If enabled, the service configuration retrieved from the local database or an external policy server is processed by the local policy script (which may change the service configuration or cause the call to be rejected).
Script	Only applies if Apply local policy is selected. Enter a jinja2 script that takes the existing service configuration (if any) and optionally modifies or overrides the service settings.
Media location policy	
Enable external media location lookup	If enabled, requests are sent to the external policy server to fetch the system location to use for media allocation.
Apply local policy	If enabled, the media location configuration retrieved from the local database or an external policy server is processed by the local policy script (which may change the media location configuration).
Script	Only applies if Apply local policy is selected. Enter a jinja2 script that takes the existing media location configuration and optionally modifies or overrides the location settings.
Directory	
Enable external directory lookup	If enabled, requests are sent to the external policy server to fetch directory information (that can be used by some Infinity Connect clients to display a phonebook).
Registration requests	
Enable external registration policy	If enabled, requests are sent to the external policy server to determine whether a device alias is allowed to register to a Conferencing Node.

3. Select **Save**.
4. Go to **Platform > Locations**.
5. Select each location in turn and specify the **Policy profile** that the Conferencing Nodes in that location should use when making policy decisions.

Pexip Infinity license installation and usage

Before you can use the Pexip Infinity platform to make calls, you must install appropriate licenses.

Note that in addition to having sufficient **licenses**, your deployment must also have sufficient **capacity** to support the number of calls being placed. For more information, see [Capacity planning](#).

Types of licenses

The terms of your Pexip Infinity license can vary according to your licensing agreement. However, your license ultimately permits the use of the Pexip Infinity software, and additionally specifies:

- the total number of [concurrent calls](#) that can be made across the entire Pexip Infinity deployment
- the total number of [Virtual Meeting Rooms and Virtual Auditoriums](#) that can be configured at any one time
- whether the [VMR Scheduling for Exchange](#) feature is enabled
- the total number of endpoints that can be used for [One-Touch Join](#)
- the total number of [Google Meet](#) access tokens you can configure for Google Meet integration

- the total number of [Microsoft Azure tenants](#) you can configure for Microsoft Teams integration
- whether [Epic telehealth profiles](#) can be configured
- and a [system](#) license that permits basic operation of the Pexip Infinity platform.

There is no limit to how many Conferencing Node servers you can deploy, or to the number of locations in which you deploy them. No special licenses are required to deploy to cloud IaaS services such as Microsoft Azure, AWS or Google Cloud Platform. Many other features and services such as registrations, call control, branding, and interoperability with Microsoft Skype for Business also do not require any additional licenses.

Licensing

Using 2 out of 200 port licenses.
Using 0 out of 20 audio licenses.
Using 78 out of 200000 vmr licenses.
Using 10 out of 12000 OTJ Endpoint licenses.

Fulfillment ID	Status	Type	Concurrent	Expiration date	Trust flags
PL-qvz13W5252d5u99-4026-2702-27464e-4793e	Enabled	port	100	12-Dec-2020	Trusted
PL-rtatba-22a7f02a-9827-4471-ea2a-11e8693a8040	Enabled	teams	1	12-Dec-2020	Trusted
PL-vmr-472a2739-1e98-4e2a-2a27-7c7f6c318e5	Enabled	vmr	100000	12-Dec-2020	Trusted
PL-audio-7a222a2a-2a50-432a-952a-5297a0ff1afe	Enabled	audio	10	12-Dec-2020	Trusted
PL-ghm-9222a273-2a2a-432a-952a-5297a0ff1afe	Enabled	ghm	1	12-Dec-2020	Trusted
PL-scheduling-242a2a27-2a2a-432a-952a-5297a0ff1afe	Enabled	scheduling	1	12-Dec-2020	Trusted
PL-otj-2a2739-1e98-4e2a-2a27-7c7f6c318e5	Enabled	otj	6000	12-Dec-2020	Trusted


8 Licenses

Examples of the different types of licenses that can be installed

Concurrent calls

Before you can place calls to Pexip Infinity services (Virtual Meeting Room, Virtual Auditoriums, Virtual Receptions and the Infinity Gateway), you must install sufficient call licenses. Two types are available:

- **port**: the total number of concurrent video calls (allows video, audio, screen share, full motion HD presentations, images and PDF content sharing). Port licenses are mandatory.
- **audio**: the total number of concurrent audio-only calls (the audio-only participant can also send and receive images and PDF presentation content). Audio licenses are optional.

 The [Live view](#) page ([Status > Live View](#)) lets you review current and historic usage charts showing a breakdown of participants by location, protocol, license type and the different conference types being hosted.

VMRs

The **vmr** license specifies the total number of Virtual Meeting Rooms plus Virtual Auditoriums that can be configured at any one time.

Scheduling

The **scheduling** license is optional and enables the VMR Scheduling for Exchange feature. If you are using this feature in your deployment, you must also have sufficient VMR licenses installed. See [VMR Scheduling for Exchange](#) for more information.

One-Touch Join

The **otj** license is optional and specifies the number of endpoints that can use One-Touch Join. See [One-Touch Join](#) for more information.

Google Meet

The **ghm** license is optional and allows you to configure Google Meet access tokens, and to provide gateway services into Google Meet conferences. The **ghm** license specifies the total number of access tokens you can configure. Note that appropriate call licenses are also required for each gateway call that is placed into a Google Meet conference.

Microsoft Teams / Azure tenants

The **teams** license is optional. It allows you to configure Microsoft Azure tenants and route calls to Microsoft Teams. The **teams** license controls how many Azure tenants can be configured; there is no limit to the number of Teams Connectors you can install or the number of instances within each Teams Connector. Multiple Teams tenants (tenant IDs) can use the same Teams Connector.

Note that appropriate call licenses are also required for each gateway call that is placed into a Microsoft Teams conference.

See [Integrating Microsoft Teams with Pexip Infinity](#) for more information.

Epic telehealth profiles

The **telehealth** license is optional and allows you to configure Epic telehealth profiles. See [Epic telehealth integration with Pexip Infinity](#) for more information.

System license

The **system** license permits basic operation of the Pexip Infinity platform and is always provided with every Pexip Infinity purchase. In the Administrator interface it may appear as a distinct license or it may alternatively be coupled with the **port** license.

Call license allocation

When a Transcoding Conferencing Node receives a request from a participant to join a conference or place a call via the Infinity Gateway, it contacts the Management Node to see whether there is an appropriate license available. If so, the call is allowed and one concurrent call license is allocated to that participant for the duration of the call. When the call is terminated, the license is returned to the pool. In general, each participant consumes a single concurrent license, and no participant can consume more than one license at a time. Specifically:

- A **port** license is consumed when making any type of video call (Full HD, HD and SD).
- If your system includes **audio** licenses:
 - a participant making an audio-only call will consume an audio license
 - if a participant attempts to make an audio-only call, but all audio licenses are currently in use, a port license is used instead, if one is available; otherwise the call is rejected due to insufficient call licenses as normal. If a port license has been used, the audio-only call uses the port license for the duration of the call, even if audio licenses become available while that call is still in progress
 - if a participant who is making an audio-only call (and therefore consuming an **audio** license) escalates that call to a video call, a **port** license is used, and the audio license is returned to the pool. Likewise, if a video call becomes an audio-only call, an audio license is used and the port license is returned to the pool.
- If your system **only** includes **port** licenses, then any type of call that normally requires an audio license will consume a port license.
- If you attempt to make a call that requires a **port** license and there are no port licenses currently available, but there is an **audio** license available:
 - the call will go ahead as an audio-only call, and consume an **audio** license
 - the participant sees an "insufficient video licenses" screen, and is seen as an audio-only participant by the other conference participants
 - the call will not automatically escalate to video if a port license becomes available.
- Infinity Connect clients and all other endpoints consume one call license per connection when sending or receiving media (audio or video). They consume a single audio or port license as appropriate.
- Infinity Connect presentation and control-only participants **do not** consume a license, even when sending slides (images or PDF), sharing their screen, or receiving presentations.
- Skype for Business / Lync participants (connected directly to a VMR) who are only sending or receiving presentations consume an audio license.
- [Gateway calls](#) consume two licenses: one for the inbound leg of the call and one for the outbound leg (each license is audio or port as appropriate).
- Any participants who are directly connected to an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet, do not consume a call license (they report a license type of "Not required").
- Proxying Edge Nodes do not affect the call licensing requirements for an endpoint connection.

If a Conferencing Node is unable to contact the Management Node, the call is permitted on the assumption that a license is available. After the Management Node has been out of contact for a grace period of 14 days, all calls will be rejected.

If there are no licenses available, or the existing license is invalid, the participant is advised that they cannot join the conference along with the reason why, and a corresponding message is written to the support log.

For more information about license usage, contact your Pexip authorized support representative.

Insufficient licenses

If your Pexip Infinity system reports **insufficient call licenses**, this means that there are valid licenses installed on the system, but at the point at which a participant tried to join a conference, all of the existing call licenses were in use. To remedy this, either wait until one or more existing conferences have completed, thus freeing up some call licenses, or add more call licenses to your system.

In some cases your licenses may include an overdraft. This is intended to cover instances where the number of concurrent calls temporarily exceeds the number of available licenses.

Note that if your system includes port licenses and audio licenses, and you attempt to make a call that requires a **port** license and there are no port licenses currently available, but there is an **audio** license available, then the call will go ahead as an audio-only call and consume an **audio** license. See [Call license allocation](#) above for more information.

Invalid license

If your Pexip Infinity reports an **invalid license**, this could mean that:

- the license has not been activated
- the existing license has expired
- the existing license has become corrupt (this could occur, for example, if the Management Node reboots after an upgrade and comes back up on a different physical blade with a new MAC address).

Licenses become valid at 00:00:00 UTC on the day they start and expire at 23:59:59 UTC on the day they expire.

Viewing existing licenses and current usage

To see the licenses that are installed on your system, and the number of licenses currently being consumed, go to **Platform > Licenses**.

The top of the **Licensing** section states, for each type of license installed, how many concurrent licenses are currently being consumed out of the total available. Underneath are listed all the licenses that have been activated successfully and how many concurrent calls/VMRs they specify. You can select a license to see further information about it.

The **Stored License Request** section lists any licenses that were not activated or returned automatically, and are awaiting [manual processing](#).

Adding licenses

Before adding or moving a license:

- You must have an active internet connection from the Management Node.
- License requests are sent to either [pexip.flexnetoperations.com](#) or [activation.pexip.com](#) (depending on the type of activation key provided). You must therefore configure your firewall to allow:
 - a connection from the Management Node to [pexip.flexnetoperations.com](#) on HTTPS port 443, and
 - a connection from the Management Node to [activation.pexip.com](#) on HTTPS port 443 (unless your Management Node has a [web proxy configured](#), in which case all licensing requests to [activation.pexip.com](#) are sent via the web proxy).

To add a new license:

1. On the Pexip Infinity Administrator interface, go to **Platform > Licenses**.
2. Select **Add License**.
3. In the **License entitlement key** field, enter the activation key provided by your Pexip authorized support representative.
4. Leave the **Manually activate** checkbox clear (unless you have been instructed to select this option by your Pexip authorized support representative, or you need to perform a manual activation due to network connectivity issues).
5. Select **Save**.

Pexip Infinity will automatically generate a file containing the license request. It then attempts to contact the Pexip licensing server and send it this file to activate the license.

- If the license is activated successfully, you are returned to the **Licensing** page and the new license is shown under the **Licensing** section.
- If the activation attempt is unsuccessful (for example, if the Management Node was unable to establish a connection to the Pexip licensing server), or you selected **Manually activate**, the license is saved as a **Stored license request**. You must then [activate it manually](#).

Moving a license (e.g. when redeploying a Management Node)

Before adding or moving a license:

- You must have an active internet connection from the Management Node.
- License requests are sent to either pexip.flexnetoperations.com or activation.pexip.com (depending on the type of activation key provided). You must therefore configure your firewall to allow:
 - a connection from the Management Node to pexip.flexnetoperations.com on HTTPS port 443, and
 - a connection from the Management Node to activation.pexip.com on HTTPS port 443 (unless your Management Node has a [web proxy configured](#), in which case all licensing requests to activation.pexip.com are sent via the web proxy).

You may need to move a license from one Management Node to another. You would typically need to do this if you are redeploying a Management Node, such as when setting up a new test or demonstration environment, or changing the Management Node's IP address (see [Moving, restoring or changing the IP address of the Management Node](#)).

- ❗ To move a license between Management Nodes, you must deactivate the existing license on your 'old' Management Node before reactivating it on the 'new' Management Node.

On the **old** Management Node:

1. Go to **Platform > Licenses**.
2. Select the license you want to deactivate.
3. Make a note of the **License entitlement key**.
4. Select **Return license**.

The system will attempt to contact the Pexip licensing server to automatically deactivate the license:

- If successful, the license is removed from the list of licenses.
- If unsuccessful, you must manually deactivate the license by selecting **Manually Return License** and then following the same steps as described in [Manually processing a stored license request \(offline activation/return\)](#).

On the **new** Management Node:

1. Go to **Platform > Licenses**.
2. Select **Add License**.
3. In the **License entitlement key** field, enter the entitlement key from the 'old' Management Node that you noted previously.
4. Select **Save**.

The system will attempt to activate the license on the 'new' Management Node:

- If the license is activated successfully, you are returned to the **Licensing** page and the new license is shown under the **Licensing** section.
- If the activation attempt is unsuccessful (for example, if the Management Node was unable to establish a connection to the Pexip licensing server), or you selected **Manually activate**, the license is saved as a **Stored license request**. You must then [activate it manually](#).

Manually processing a stored license request (offline activation/return)

To manually process a stored license request:

1. Go to **Platform > Licenses**.
The **Stored License Request** section lists the licenses that could not be activated or returned automatically, and are awaiting manual activation/return.
2. Select the license request you want to process.
3. Select **Export stored license request**.
This generates a file containing the request.
4. Download the request file and send it to your Pexip authorized support representative for activation/return. They will respond with a fulfillment file.
5. When you receive the fulfillment file, go back to **Platform > Licenses**, select the stored license request, and select **Complete stored license request**.
6. Browse to the location of the fulfillment file you have received.
7. Select **Save**.

The license should now be activated and appear in the **Licensing** section, or returned as appropriate.

Repairing a license

If a license becomes corrupt (for example if the Management Node MAC address has changed), a **Repair trusted license storage** button will appear under the license information. To reactivate the license, select **Repair trusted license storage**. This will normally resolve the issue automatically; in some circumstances the repair operation will result in a stored license request, in which case follow the procedure above for [Manually processing a stored license request \(offline activation/return\)](#). Note that a repair operation creates a single request, regardless of the number of licenses that are corrupt.

Content of the license request file

The same license request file is used for both automatic and manual license activation/return. The content of this file includes:

- the license request information (i.e. the detail of what is being requested)
- opaque machine identifiers (computed from various information sources, but not exposing the actual values).

Managing TLS and trusted CA certificates

TLS certificates are used by the Management Node and each Conferencing Node to verify their identity to clients connecting to them over HTTPS (web) or SIP TLS. These clients include:

- video endpoints
- web browsers (including the Infinity Connect web app)
- Infinity Connect mobile clients (certificates are mandatory for these clients)
- third-party video network infrastructure devices
- Outlook clients (if the VMR Scheduling for Exchange service is enabled).

You can use Pexip Infinity's inbuilt [Certificate Signing Request \(CSR\) generator](#) to assist in acquiring a server certificate from a Certificate Authority.

Further information:

- To configure Pexip Infinity to verify peer certificates, and mutual TLS authentication, see [Verifying SIP TLS connections with peer systems](#).

Note that communication between the Management Node and Conferencing Nodes, and between Conferencing Nodes themselves, does not rely on TLS certificates; instead it uses an IPsec transport. For more information see [Encryption methodologies](#).

Certificates overview

The Public Key Infrastructure (PKI) provides a framework for digital certificate management. It provides the following benefits:

- **Authentication:** identities are validated to ensure that only authorized users and devices have access to a server.
- **Encryption:** sessions can be encrypted, so information can be transmitted privately.
- **Data integrity:** ensures that any messages or data transferred to and from devices and servers is not altered.

The primary elements of the PKI are:

- **Public/private key pairs:** public and private keys are used to encrypt and decrypt the information being transferred to a server. Only the private key, which is kept secret by the server, can decrypt the information that is encrypted by the public key. This mechanism is known as asymmetric cryptography (as the encryption is done using non-identical keys); the two keys are mathematically related, and whatever is encrypted with a public key can only be decrypted by its corresponding private key and vice versa.
- **Certificate:** a certificate contains the public key and information about its owner (often referred to as the subject and is typically expressed as a hostname or domain name) and its issuer (typically a trusted, third-party Certificate Authority). This certificate metadata is formatted according to the ITU-T X.509 international standard. A certificate is not considered valid unless it has been directly or indirectly signed by a trusted CA.
- **Certificate Authority (CA):** a CA is an organization such as Symantec or Comodo that issues digital certificates to corporations after having verified the applicant's identity. The certificate is proof that a certain server or website is owned by a certain organization. A CA can use its own private key to sign the certificates it issues. That signed certificate can then be verified as being signed by a trusted CA by checking the signature against the CA's own root certificate (via its public key). However, many CAs do not sign with their root certificate, but instead with an intermediate certificate — an intermediate authority is a certificate issuer that has itself been issued by a root or another higher level intermediate authority. This method creates a chain of trust.
- **Chain of trust:** when a device validates a certificate, it compares the certificate issuer with its list of trusted CAs. If a match is not found, it checks if the certificate of the issuing CA was issued by a trusted CA, and so on until the end of the certificate chain. The top of the chain, the root certificate, must be issued by a trusted Certificate Authority. Web browsers and other clients typically have a list of CA certificates that they trust, and can thus verify the certificates of individual servers, however, the server is often required to present a full certificate chain along with its server certificate.

TLS certificate usage within Pexip Infinity

The certificates used within Pexip Infinity are standard digital certificates, but they are referred to as TLS certificates as they are used when establishing a TLS connection to a Pexip node.

The clients that connect to Pexip Infinity over TLS must trust the identity of the Certificate Authority (CA) that signed the node's TLS certificate. The Pexip Infinity platform ships with a self-signed certificate for the Management Node, and each Conferencing Node is deployed with a self-signed certificate. These certificates have a 4096 bit public key and are also appended with 2048 bit Diffie-Hellman parameters. Each self-signed certificate will expire after 30 days.

As these certificates are self-signed, they will not be trusted by clients. We therefore recommend that you replace these certificates with your own certificates that have been signed by either an external CA or a trusted internal CA.

You can use a tool such as <https://www.sslshopper.com/ssl-checker.html> to verify certificates and the chain of trust (specify port 5061 i.e. use the format <domain>:5061 for the server hostname to ensure that SIP TLS connections are checked).

The Management Node and Conferencing Nodes enable HSTS (HTTP Strict Transport Security) to ensure greater security. This means that if your deployment moves from using a valid TLS certificate to using an invalid certificate (e.g. you redeploy a Conferencing Node, or your certificate expires or is invalidated for some reason) then certain web browsers will stop you from accessing that node via the web when using the DNS name of that node, until you correct the certificate issue. You may browse directly to the IP address of the node in the meantime.

In general, to achieve encrypted communication using TLS the following must happen:

1. The CA issues a signed certificate which is uploaded to the server.
2. When a client needs to communicate with the server, it sends a request to the server asking it to provide identification.
3. The server sends back a copy of its TLS certificate and its public key.
4. The client checks whether the CA that issued the certificate is one that it trusts.
5. If the CA is trusted, and if the certificate is otherwise valid, the client creates a session key encrypted with the server's public key and sends it to the server.
6. The server decrypts the session key. It then uses the session key to encrypt an acknowledgment which it sends to the client in order to initiate the encrypted communication.
7. The server and the client now encrypt all communication using the session key.

Certificate usage guidelines

When requesting/generating certificates for your Pexip Infinity platform:

- Do not use SHA-1 certificates on your Conferencing Nodes — use SHA-256 certificates instead. (Some clients, such as iOS devices, already mandate the use of SHA-256 certificates, and browsers will soon stop accepting SHA-1 certificates.)
 - If browsers (Infinity Connect web app clients, for example) need to access your Pexip Infinity platform, ensure that your certificates contain at least one subject alternative name (SAN) entry — typically a repeat of the Common Name. (Chrome and Firefox browsers are dropping support for Common Name matching.)
- i** Wildcard TLS certificates are not supported in SIP or Microsoft Skype for Business / Lync environments (as per RFC 5922). If you are using SIP or Skype for Business / Lync, your Conferencing Nodes must not use wildcard TLS certificates.

Alarms

An alarm is raised on the Management Node if:

- the Management Node or a Conferencing Node has no associated TLS certificate
- a TLS certificate has an incomplete chain of trust to the root CA certificate
- one or more of your trusted CA certificates or TLS certificates is due to expire within the next 30 days.

Managing a node's TLS server certificate

You can [upload](#), [view/modify](#), [delete](#) and [download](#) the TLS server certificates that are used by the Management Node and by each Conferencing Node. You can also generate a [certificate signing request](#) for an existing certificate / subject name.


Note that after making any changes to certificates, you need to wait for the files to be synchronized to the relevant Management Node or Conferencing Node (typically after approximately one minute). If changing the certificates in a chain, a reboot of the associated Conferencing Nodes may be required if the changes do not produce the desired effect.

Uploading a TLS server certificate


To upload a new TLS server certificate for the Management Node or a Conferencing Node:

1. From the Pexip Infinity Administrator interface, go to **Platform > TLS Certificates**.
2. Select **Add TLS certificate**.

3. Complete the following fields:

TLS certificate	<p>Paste the PEM-formatted certificate into the text area or alternatively select the file containing the new TLS certificate.</p> <p> You must upload the certificate file that you have obtained from the Certificate Authority (typically with a .CRT or .PEM extension). Do not upload your certificate signing request (.CSR file).</p> <p>The certificate must be valid for the DNS hostname or FQDN of the Management Node or Conferencing Node to which it will be assigned.</p> <p>You can paste multiple certificates into the text area, but one of those certificates must pair with the associated private key.</p>
Private key	<p>Paste the PEM-formatted private key into the text area or alternatively select the file containing the private key that is associated with the new TLS certificate.</p> <p>Private key files typically have a .KEY or .PEM extension. Pexip Infinity supports RSA and ECDSA keys.</p>
Private key passphrase	If the private key is encrypted, you must also supply the associated passphrase.
TLS parameters	<p>Optionally, paste any additional PEM-formatted parameters into the text area or alternatively select the file containing the parameters that are to be associated with the new TLS certificate.</p> <p>Custom DH parameters and an EC curve name for ephemeral keys can be added. Such parameters can be generated through the OpenSSL toolkit using the commands <code>openssl dhparam</code> and <code>openssl ecparam</code>. For example, the command <code>openssl dhparam -2 -outform PEM 2048</code> generates 2048 bit DH parameters.</p> <p>Note that these parameters can alternatively be added 'as is' to the end of the TLS certificate.</p>
Nodes	<p>Select one or more nodes to which the new TLS certificate is to be applied.</p> <p>If required, you can upload a certificate and then apply it to a node later.</p>

4. Select **Save**.

-  If a certificate with the same subject name already exists (e.g. when replacing an expired certificate), the new certificate is uploaded alongside the original certificate (unless the issuer and serial number details are identical, in which case the existing certificate is updated with the new contents from the file). If the original TLS certificate was assigned to one or more Conferencing Nodes you need to move those node assignments over to the new certificate.

Viewing or modifying existing TLS certificates and changing node assignments

To view information about an existing TLS certificate, change a certificate's contents, or change the nodes to which a certificate is applied:

1. Go to **Platform > TLS Certificates**.

By default you are shown a list and the current status of **All certificates** that have been uploaded. Status values include:

- **Good:** it is a good, valid certificate.
- **Temporary:** it is a self-signed certificate.
- **Weak signature:** the certificate is signed with SHA-1 or another old signature. We recommend that you use SHA-256 certificates instead.
- **Empty subject:** the certificate only has SANs. It is a valid certificate but many browsers will not accept it.
- **Over <n> days:** all browsers will not trust certificates that are valid for longer than 825 days, and Safari will also not trust certificates with a start date on or after 1st September 2020 and that are valid for longer than 398 days.
- **<n> days left:** when the certificate is due to expire within the next 60 days.
- **Expired:** when the certificate has expired.

You can alternatively select to view **Certificates by Node** to see which certificate has been assigned to the Management Node or to a particular Conferencing Node.

2. Select the subject name of the certificate you want to view or modify.

The decoded certificate data is shown, including any chain of trust information.

3. If required, you can modify the:

- **Nodes** to which the certificate is assigned. If you assign a certificate to a node, it will automatically replace any other certificate that was previously assigned to that node.
- **TLS certificate data** (by expanding the **PEM-formatted data** section).
- **TLS parameters** associated with the certificate (by expanding the **PEM-formatted data** section).

You cannot modify the private key.

4. Select **Save**.

Uploading multiple TLS certificate files

To upload a batch of TLS certificates:

1. Go to **Platform > TLS Certificates**.
2. Select **Import files**.
3. Select **Choose Files** to pick one or more PEM-formatted text files that you want to import.
 - The files should contain server TLS certificates with matching private keys.
 - Private keys can be uploaded as separate files or appended to the server TLS certificate file(s).
 - DH or EC parameters may be appended to each server TLS certificate.

Note that trusted CA certificate files can also be imported via this method if required.

4. Enter the **Private key passphrase** if the private key is encrypted.
 5. Select **Import**.
This adds the certificates in the selected files to the existing list of TLS certificates (or to the list of trusted CA certificates, if appropriate).
 6. You can then select each imported TLS certificate in turn and assign it to a Management Node or one or more Conferencing Nodes as appropriate.
- i** If a certificate with the same subject name already exists (e.g. when replacing an expired certificate), the new certificate is uploaded alongside the original certificate (unless the issuer and serial number details are identical, in which case the existing certificate is updated with the new contents from the file). If the original TLS certificate was assigned to one or more Conferencing Nodes you need to move those node assignments over to the new certificate.

Deleting an existing TLS certificate

To delete one or more TLS certificates:

1. Go to **Platform > TLS Certificates**.
2. Select the boxes next to the certificates to be deleted, and from the **Action** drop-down menu select **Delete selected TLS certificates** and select **Go**.

Downloading an existing TLS certificate

To download an existing TLS certificate:

1. Go to **Platform > TLS Certificates**.
2. Select the subject name of the certificate you want to download.
The certificate data is shown.
3. Go to the bottom of the page and select **Download**.
4. By default the **certificate** itself and any **intermediate certificates** are selected to be included in the download.
You can also choose to include the **private key** (in which case you must also supply a **passphrase**).
5. Select the download format, either **PEM** or **PKCS # 12 (PFX)**.
6. Select **Download**.

A file called **<subject_name>_certificate** or **<subject_name>_keycert** (if the private key was included) with either a **.pem** or **.pfx** extension is downloaded. This file contains the certificate and/or the private key as requested.

You can also download multiple certificates into one file: select the boxes next to the certificates to be downloaded, and from the **Action** drop-down menu select **Download** and select **Go**. In this case the generated file is called **all_certificates** or **all_keycerts** (if private keys were included).

Note that you cannot download a temporary / self-signed certificate.

- i** You can use the Pexip Infinity Management Node to convert PEM certificates to PFX format (or vice versa), by uploading a PEM-formatted certificate and then downloading it again in PFX format. When downloading you can also include the private key and all necessary intermediate certificates in the PFX bundle.

Replacing an existing certificate (by generating a new certificate signing request)

You can generate a certificate signing request (CSR) for an existing certificate / subject name, for example if your current certificate is soon due to expire and you want to replace it. Before generating the CSR you can change the certificate data to be included in the new request, such as adding extra subject alternative names (SANs) to those already present in the existing certificate.

For instructions on how to do this, see [Requesting a certificate signing request \(CSR\) for an existing certificate / subject name](#).

Managing trusted CA certificates

Trusted CA certificates are used within Pexip Infinity to:

- verify client certificates presented to Pexip Infinity when [SIP TLS verification mode](#) is enabled
- provide a certificate [chain of trust](#) when clients connect to a Conferencing Node over SIP TLS
- verify the server certificate on a video network infrastructure device when a Conferencing Node makes a SIP TLS outbound connection to that device, and that device chooses a cipher suite that requires authentication and a certificate is exchanged
- verify connections to an LDAP server
- verify connections to an Exchange on-premises server when using [VMR Scheduling for Exchange](#) or [One-Touch Join](#).
- verify connections to an endpoint's API, if required, when using One-Touch Join.

Pexip Infinity ships with an inbuilt list of trusted CA certificates. This list is based on the [Mozilla CA Certificate Store](#) and cannot be modified.

In addition, you can [upload a customized set](#) of trusted CA certificates to the Pexip Infinity platform.

Chain of trust

For a server's TLS certificate to be trusted by a client, the client must be configured to trust the Certificate Authority (CA) that signed the server certificate. Many CAs do not sign with their root certificate, but instead with an intermediate certificate. Clients, however, may only trust the root CA. Therefore the server (in this case the Management Node or Conferencing Node) is often required to present a full certificate chain along with their TLS server certificate.

When this is the case, the chain of intermediate CA certificates must be installed on the Management Node to ensure that the certificate chain of trust is properly established when clients connect to a Conferencing Node over SIP TLS.

To do this you must upload a custom **Trusted CA certificates** file that contains all the required CA certificates, one after each other, in PEM format.

Uploading and managing additional trusted CA certificates

You can upload a customized set of trusted CA certificates to the Pexip Infinity platform. Any trusted CA certificates uploaded here are used in addition to the default set of trusted CA certificates that ships with Pexip Infinity.

To manage the set of custom trusted CA certificates, go to **Platform > Trusted CA Certificates**. This shows a list and the current status of all the trusted CA certificates that have been uploaded. From here you can:

- **Upload a file of Trusted CA certificates:** select **Import files**, select **Choose Files** to pick one or more PEM files that you want to import, and then select **Import**.
This adds the certificates in the selected files to the existing list of trusted CA certificates (or to the list of TLS certificates, depending on the certificate types contained in the file). If a certificate with the same subject name already exists (e.g. when replacing an expired certificate), the new certificate is uploaded alongside the original certificate (unless the issuer and serial number details are identical, in which case the existing certificate is updated with the new contents from the file).
- **View or modify an existing certificate:** select the **Subject name** of the certificate you want to view. The decoded certificate data is shown.
If required, you can modify the PEM-formatted certificate data and select **Save**.
- **Download all certificates:** select **Export**. A **ca-certificates.pem** file containing all of the custom-added certificates in PEM format is created and automatically saved to your local file system.

- **Delete one or more certificates:** select the boxes next to the certificates to be deleted, and from the Action drop-down menu select *Delete selected Trusted CA certificates* and select Go.

Certificate signing requests (CSRs)

To acquire a server certificate from a Certificate Authority (CA), a certificate signing request (CSR) has to be created and submitted to the CA. You can generate a CSR from within Pexip Infinity, and then upload the returned certificate associated with that request.

You can create a new CSR for any given subject name / node, or if you have an existing certificate already installed on a Pexip Infinity node that you need to replace (for example if it is due to expire) you can create a CSR based on the existing certificate data.

CSRs generated via Pexip Infinity always request client certificate and server certificate capabilities.

This topic covers:

- [Requesting a certificate signing request \(CSR\) for an existing certificate / subject name](#)
- [Creating a new certificate signing request](#)
- [Uploading the signed certificate associated with a certificate signing request](#)
- [Troubleshooting](#)
- [Modifying a CSR](#)

Requesting a certificate signing request (CSR) for an existing certificate / subject name

You can generate a certificate signing request (CSR) for an existing certificate / subject name, for example if your current certificate is soon due to expire and you want to replace it. Before generating the CSR you can change the certificate data to be included in the new request, such as adding extra subject alternative names (SANs) to those already present in the existing certificate.

To generate a CSR for an existing certificate / subject name:

1. Go to **Platform > TLS Certificates**.
2. Select the subject name of the certificate for which you want to generate a CSR.
The certificate data is shown.
3. Go to the bottom of the page and select **Create certificate signing request**.
You are taken to the Add Certificate signing request page, and the CSR data is defaulted to the contents of the certificate you selected.
4. If required you can change the certificate data, such as the subject alternative names (SANs) and subject fields.
Note that you cannot change the private key — the CSR uses the same private key as the original certificate.
5. Select **Save**.
The CSR is generated and you are taken to the **Change Certificate signing request** page.
6. Select **Download**.
This downloads the CSR to your local file system, with a filename in the format **<subject-name>.csr**.
Note that the private key is not downloaded, or included within the CSR.
7. You can now submit this CSR file to your chosen CA for signing.
The CA will then send you a signed certificate which you can upload into Pexip Infinity (see [Uploading the signed certificate associated with a certificate signing request](#)).

Note that:

- The validity, expiry date etc. of the existing certificate is not affected when you create a CSR to replace it.
- You cannot generate a CSR for an existing temporary / self-signed certificate.
- If the CSR generation fails with a "It was not possible to automatically create a certificate signing request from this certificate" message, then there was a problem with validating the original certificate data, most likely an invalid subject name or an invalid country code. In this case you will have to create the CSR manually.

Creating a new certificate signing request

To generate a CSR within Pexip Infinity:

1. Go to **Utilities > Certificate Signing Requests**.
2. Select **Add Certificate signing request**.
3. Complete the following fields:

TLS Certificate	<p>Create non-renewal CSR is selected by default. This lets you create a new CSR.</p> <p>To create a renewal CSR based on an existing certificate, choose a different subject name / issuer from the list (in which case the subject name and private key fields below are not displayed).</p>
Subject name	<p>Select the name to be specified as the Common Name field of the requested certificate's subject. This is typically set to the FQDN of the node on which the certificate is to be installed.</p> <p>The available options are prepopulated with the FQDNs (hostname plus domain) of the Management Node and each currently deployed Conferencing Node. The list also includes any SIP TLS FQDN names of your Conferencing Nodes, if such names have been configured and are different from the node's FQDN.</p> <p>If you want to specify a custom Common Name instead, select User-provided custom Common Name.</p>
Custom subject name	Enter the name that you want to use as the Common Name field of the requested certificate's subject, if you have selected User-provided custom Common Name above.
Private key type	<p>Select the type of private key to generate, or select Upload user-provided private key if you want to provide your own private key.</p> <p>Default: RSA (2048bit)</p>
Private key	<p>Only applies if you have selected Upload user-provided private key above.</p> <p>Enter the PEM formatted RSA or ECC private key to use when generating your CSR. You can either paste the key into the input field or upload the private key file from your local file system.</p>
Private key passphrase	<p>Only applies if you have selected Upload user-provided private key above.</p> <p>If the private key is encrypted, you must also supply the associated passphrase.</p>
Subject alternative names	<p>Select the subject alternative names (SANs) to be included in the CSR. This allows the certificate to be used to secure a server with multiple names (such as a different DNS name), or to secure multiple servers using the same certificate.</p> <p>You can choose from the same list of names presented in the Subject name field. Note that the name you choose as the Common Name is automatically included in the generated CSR's list of SANs (even if you remove it from the Subject alternative names list shown here).</p> <p>In some deployments it may be more practical to generate a single CSR in which all of your Conferencing Node FQDNs are included in the list of SANs. This means that the same single server certificate returned by the CA can then be assigned to every Conferencing Node.</p> <p>When integrating with Microsoft Skype for Business / Lync, SAN entries must be included for every individual Conferencing Node in the public DMZ (public DMZ deployments) or in the trusted application pool (on-prem deployments).</p>
Additional subject alternative names	<p>Optionally, enter a comma-separated list of additional subject alternative names to include in the CSR. For example:</p> <ul style="list-style-type: none"> ◦ When receiving SIP or Skype for Business / Lync (MS-SIP) calls, the certificate on the Conferencing Node receiving the call should include the domain names (e.g. vc.example.com) that are used in any DNS SRV records that are used to route calls to those Conferencing Nodes. ◦ When integrating with on-prem Skype for Business / Lync deployments you would typically need to add the trusted application pool FQDN.
Additional subject fields (if required you can enter the following additional CSR attributes; these are all blank by default)	
Organization name	The name of your organization.

Department	The department within your organization.
City	The city where your organization is located.
State or Province	The state or province where your organization is located.
Country	The 2 letter code of the country where your organization is located.
Advanced (in most scenarios you should leave the advanced options to their default settings)	
Include Microsoft certificate template extension	Select this option to specify a (Microsoft-specific) certificate template in the CSR. This is needed when using the Certification Authority MMC snap-in to request a certificate from an enterprise CA. Selecting this option causes the 'WebServer' certificate template to be specified. Default: disabled.
Include Common Name in Subject Alternative Names	Specifies whether to include the requested subject Common Name in the Subject Alternative Name field of the CSR. Default: enabled.

4. Select **Save**.

You are taken to the **Change Certificate signing request** page.

5. Select **Download**.

This downloads the CSR to your local file system, with a filename in the format <subject-name>.csr.

Note that the private key is not downloaded, or included within the CSR.

6. You can now submit this CSR file to your chosen CA for signing.

The CA will then send you a signed certificate which you can upload into Pexip Infinity (see below).

Uploading the signed certificate associated with a certificate signing request

When the Certificate Authority sends you a signed certificate in response to your CSR, you can upload that certificate into Pexip Infinity and assign it to one or more of your nodes. Make sure that you upload it via the **Certificate Signing Requests** page as this ensures that it is linked with the private key associated with your original CSR.

To upload the signed certificate:

1. Go to **Utilities > Certificate Signing Requests**.

2. Select the original CSR that is associated with the signed certificate.

You are taken to the **Change Certificate signing request** page.

3. In the **Certificate** field either paste the PEM-formatted certificate into the input field or upload the certificate file from your local file system.

The certificate file that you have obtained from the Certificate Authority typically has a .CRT or .PEM extension. Do not upload your certificate signing request (.CSR file).

4. Select **Complete**.

Providing it is a valid certificate and is based on the original CSR:

- the certificate is uploaded and automatically linked with the private key associated with your original CSR.
- if you are uploading a replacement certificate (same subject name and private key) it will replace the existing certificate and maintain any existing node assignments.
- the original CSR is deleted.
- you are taken to the **Change TLS Certificate** page.

5. You can now assign that certificate to the Management Node or one of more Conferencing Nodes as required:

- a. From within the **Change TLS Certificate** page go to the **Nodes** field and from the **Available Nodes** list, select the nodes to which you want to assign the certificate and move them into the **Chosen Nodes** list.
- b. Go to the bottom of the page and select **Save**.

Troubleshooting

This section describes some of the error messages you may see when attempting to upload a signed certificate.

Error message	Possible cause	Resolution
Certificate and private key do not appear to be part of the same key pair	This most likely means that you have tried to upload the certificate against the wrong CSR.	Select the correct CSR and try again.

Modifying a CSR

After a CSR has been created it cannot be modified — the only available actions are to download it (for sending to a CA), or to apply the returned, signed certificate that is associated with that request.

If you need to change the content of a CSR, you should delete the original CSR and create a new CSR with the correct content.

Note that a CSR is automatically deleted when the resulting signed certificate is uploaded.

Verifying SIP TLS connections with peer systems

When a system, such as an endpoint or a video network infrastructure device, attempts to connect to a Pexip Infinity Conferencing Node, the Pexip Infinity platform can be configured to verify whether the certificate presented by the peer system is valid before the connection is allowed.

In addition, if the peer system is also configured to perform TLS verification checking (mutual TLS authentication), then that external system will verify that the certificate on the Conferencing Node is valid and has the appropriate client authentication capabilities.

Verifying peer certificates for SIP TLS connections

For connections over SIP TLS, you can use the **SIP TLS verification mode** setting to control whether the certificate presented by the peer is verified before the connection to a Pexip Infinity Conferencing Node is allowed. When this setting is enabled:

- The peer certificate is verified (in date, issued by a trusted authority etc).
- OCSP, if enabled, is used to check that the certificate has not been revoked (see [Using OCSP to check the status of certificates](#) below for more information).
- The identity of the peer as presented in the certificate is checked against the identity expected by Pexip Infinity.
- The peer certificate must be a **client** certificate (see [Mutual TLS authentication and client/server certificates](#) below for more information) — otherwise you will get "unsupported certificate" errors in the support log.

When this setting is enabled, all peers (including endpoints connecting directly with Pexip Infinity) must have their own certificate.

To enable or disable this setting, go to **Platform > Global Settings > Security**. The options available are:

Field	Description
SIP TLS verification mode	Determines whether to verify the peer certificate for connections over SIP TLS. <i>Off</i> : the peer certificate is not verified; all connections are allowed. <i>On</i> : the peer certificate is verified, and the peer's remote identities (according to RFC5922) are compared against the Application Unique String (AUS) identified by Pexip Infinity — the SIP URI — before the connection is allowed. Default: <i>Off</i> .

Using OCSP to check the status of certificates

Pexip Infinity allows you to use Online Certificate Status Protocol (OCSP) to check whether a certificate has been revoked. TLS certificates that support OCSP are encoded with the URL of an OCSP responder — a server that will check and respond with the status of the certificate.

OCSP checking only applies when **SIP TLS verification mode** is *On*. To enable or disable the use of OCSP, and to configure the URL of the OCSP responder, go to **Platform > Global Settings > Security**. The options available are:

Field	Description
OCSP state	<p>Determines whether OCSP is used to check the status of TLS certificates.</p> <p><i>Off</i>: OCSP is not used.</p> <p><i>On</i>: OCSP is used, and the request is sent to the URL specified in the TLS certificate. If no URL is specified in the TLS certificate, the OCSP responder URL configured below is used.</p> <p><i>Override</i>: OCSP is used, and the request is sent to the OCSP responder URL specified in the OCSP responder URL field, regardless of any URL encoded in the TLS certificate.</p> <p>Default: <i>Off</i>.</p>
OCSP responder URL	<p>The URL to which OCSP requests are sent if either:</p> <ul style="list-style-type: none"> the OCSP state is set to <i>On</i> but no URL is present in the TLS certificate, or the OCSP state is set to <i>Override</i> (in which case any URL present in the certificate is ignored).

Mutual TLS authentication and client/server certificates

If a Conferencing Node makes an outbound connection to, or receives an incoming connection from, a video network infrastructure device (such as a Cisco VCS) that is configured to perform TLS verification checking, then that external system will verify that the certificate on the Conferencing Node is valid. In these cases, both the Conferencing Node and the external system can adopt the role of a client as well as acting as a server. Therefore, for mutual TLS authentication — where both parties are verifying the other party's certificate — both the Conferencing Node's TLS server certificate and the external system's certificate must be capable of being used as a **client** certificate as well as a **server** certificate.

A certificate's capabilities are contained in its Enhanced Key Usage properties. They indicate if the certificate can be used for server authentication (typically shown as "TLS Web Server Authentication") and for client authentication (typically shown as "TLS Web Client Authentication").

When requesting certificates for your Conferencing Nodes, if you want the node to be able to verify itself as a client when connecting to an external system, then you must request that the certificate can act as a client certificate (as well as a server certificate). All CSRs generated via Pexip Infinity's inbuilt [CSR generator](#) always request client certificate and server certificate capabilities.

If you create your certificate signing requests via the OpenSSL toolkit, then you can modify the `[v3_req]` section of your openssl request configuration file so that it contains `extendedKeyUsage=serverAuth, clientAuth`. Other Certificate Authorities have different methods of requesting client authentication.

The following table summarizes the certificate requirements for inbound connections to, and outbound connections from Conferencing Nodes:

	Inbound SIP TLS connections	Outbound SIP TLS connections
When SIP TLS verification mode is <i>Off</i>	<ul style="list-style-type: none"> The Conferencing Node's certificate must have server authentication properties. 	<ul style="list-style-type: none"> Pexip Infinity enables ADH ciphers (Anonymous Diffie-Hellman) — this enables support for endpoints that do not have certificates. Pexip Infinity may or may not authenticate the far end, depending on which cipher suite is selected by the far end: <ul style="list-style-type: none"> if an ADH cipher is selected, the far end is not authenticated and certificates are not exchanged if a cipher that requires authentication is selected and a certificate is exchanged, that certificate must be valid (issued by a trusted authority, in date etc.) and it must be a server certificate.

	Inbound SIP TLS connections	Outbound SIP TLS connections
When SIP TLS verification mode is <i>On</i>	As above, plus: <ul style="list-style-type: none"> The client system/endpoint must present a valid certificate (issued by a trusted authority, in date etc.) and it must be a client certificate. The identity of the peer as presented in the certificate must match the identity expected by Pexip Infinity. 	As above, plus: <ul style="list-style-type: none"> The identity of the peer as presented in the certificate must match the identity expected by Pexip Infinity. This means that Pexip Infinity cannot connect out over TLS to endpoints that don't have a certificate (even if an ADH cipher is selected) — you must use TCP/UDP instead.
When OCSP state is <i>On</i> or <i>Override</i> (and SIP TLS verification mode is <i>On</i>)	<ul style="list-style-type: none"> The client certificate must not be revoked. 	<ul style="list-style-type: none"> The far end certificate must not be revoked.
If the far end system performs a TLS verification process	<ul style="list-style-type: none"> The Conferencing Node's certificate must have client authentication properties and must not be a self-signed certificate. 	<ul style="list-style-type: none"> The Conferencing Node's certificate must have client authentication properties and must not be a self-signed certificate.

Integrating with external systems

When integrating with external systems, Pexip Infinity may be challenged for authentication credentials, or in some cases communications from an external conferencing system may need to be authenticated by Pexip Infinity.

Managing SIP credentials

When sending a SIP request, Pexip Infinity may be challenged for authentication credentials by the host system or proxy.

The authentication challenge will include a realm that identifies the credentials that Pexip Infinity should use to respond to the challenge. The credentials to use will be provided separately by the administrator of the host system or proxy.

All realms, usernames and passwords are case sensitive.

To configure the SIP credentials used by Pexip Infinity for each realm, go to **Call Control > SIP Credentials**. The options are:

Option	Description
Realm	The realm defines the protection space of the host or proxy (such as its name or domain, e.g. sipproxy.example.com) that is challenging Pexip Infinity for authentication.
Username and Password	The username and password presented by Pexip Infinity when responding to the authentication challenge for the given realm.

CUCM Ad Hoc Conferencing credentials

In many video conferencing deployments, Pexip Infinity will be integrated with other third-party systems and products. An example is Cisco Unified Communications Manager (CUCM), which can be integrated with Pexip Infinity so that when users initiate the **Ad Hoc Conferencing** feature on supported endpoints, the resulting conference will be hosted on Pexip Infinity.

This type of integration requires that communications from the external system are authenticated by Pexip Infinity. (Note that the credentials for this authentication are distinct from those used to access the Pexip Infinity management API.)

Some external systems (such as CUCM 8.6.2) only support access over HTTP rather than the secure HTTPS protocol, so you have the option to enable this if required.

To configure Pexip Infinity with the authentication credentials and protocol to be used by systems which do not use the management API, go to **Platform > Global Settings > External System Integration**. The options available are:

Field	Description
Enable HTTP access for external systems	Access for external systems is over HTTPS by default. If this box is selected, access over HTTP is also permitted.
External system username	The username presented to Pexip Infinity by external systems attempting to authenticate with it.
External system password	The password presented to Pexip Infinity by external systems attempting to authenticate with it.

Enabling and disabling SIP, H.323, WebRTC and RTMP

All call protocols, except for SIP over UDP, are enabled by default in your Pexip Infinity deployment.

If your deployment does not include any endpoints that support a particular protocol, for security reasons you may want to disable support for those protocols across your entire Pexip Infinity deployment.


SIP over UDP is disabled by default so as to reduce the impact of SIP spam which typically uses UDP.

The configurable protocols are:

- SIP over TCP and TLS, including MS-SIP (Skype for Business / Lync)
- SIP over UDP (for incoming calls)
- H.323
- WebRTC
- RTMP

WebRTC calls can originate from the Infinity Connect desktop client, the Infinity Connect web app via Google Chrome, Microsoft Edge, Firefox, Opera and Safari (version 11 onwards) browsers, and the Infinity Connect mobile client.

RTMP and **RTMPS** (for encrypted RTMP) are used to send conference content to streaming and recording services. RTMP authentication is supported; in this case credentials are included in the URI using the syntax `rtmps://username:password@host/...`

 The [Live view](#) page (**Status > Live View**) lets you review current and historic usage charts showing a breakdown of participants by location, protocol, license type and the different conference types being hosted.

To enable or disable a particular call protocol across your Pexip Infinity deployment:

1. Go to **Platform > Global Settings > Connectivity**.
2. Select or clear the following checkboxes, as appropriate:
 - **Enable SIP** — this affects both SIP and MS-SIP (Skype for Business / Lync) over TCP and TLS
 - **Enable SIP UDP** — this affects incoming calls only over SIP UDP
 - **Enable H.323**
 - **Enable WebRTC**
 - **Enable RTMP**
3. If you have Enabled WebRTC or Enabled RTMP and wish to use the Infinity Connect clients or client API, you must also **Enable support for Pexip Infinity Connect clients and Client API**.

Break-in resistance settings to mitigate rogue calls

Common attacks on videoconferencing systems include rogue calls — such as Spam Over Internet Telephony (SPIT) or toll fraud call attempts — that are targeted at an organization's SIP (or, more rarely, H.323) infrastructure. Typically the attacker will place a large volume of calls to numeric aliases (usually using SIP UDP) to try and gain access to a VoIP to PSTN gateway — and, if successful, use the gateway to commit toll fraud.

To mitigate such attacks, the Pexip Infinity platform enables **PIN brute force resistance** and **VOIP scanner resistance** by default. If required you can disable these settings either at a global platform level, or enable/disable protection for specific locations. You can also specify an allowed set of trusted IP addresses that are exempt from the break-in checks.

These break-in resistance settings form part of a broader strategy for protecting your system; for more information see [Security best practices](#).

This topic covers:

- [Alerting the administrator to break-in attempts](#)
- [PIN brute force resistance](#)
- [VOIP scanner resistance](#)
- [Configuring the allow list of IP addresses](#)
- [Break-in prevention policy example log messages](#)

Alerting the administrator to break-in attempts

When break-in resistance protection has been triggered, an [alarm](#) is raised on the Management Node, providing information such as the source IP address of the attack and the associated Conferencing Node. The alarm remains active for the duration of the temporary block, after which time it is lowered automatically. To monitor whether break-in resistance has been triggered in the past, you can review the alarm history, or you can review the administrator log by searching for the relevant break-in policy prevention [messages](#).

PIN brute force resistance

When PIN brute force resistance is enabled, Pexip Infinity will temporarily block all access to a VMR that receives a significant number of incorrect PIN entry attempts (and thus may perhaps be under attack from a malicious actor). It blocks **all** new access attempts to a VMR for up to 10 minutes if more than 20 incorrect PIN entry attempts are made against that VMR in a 10 minute window (you can configure the number of allowed incorrect attempts, but you cannot change the time window). While blocked, it appears to any callers as though the VMR/alias does not exist any longer. There is also a [corresponding alarm](#) raised on the Management Node.

Note that this provides a measure of resistance against PIN cracking attacks, but it is not a substitute for having a long PIN (6 digits or longer recommended) and it does not protect against a determined and patient — or lucky — attacker. Also, enabling this feature could potentially allow a malicious attacker or a legitimate user with incorrect access details to prevent legitimate access to VMRs or other call services for a period.

To configure PIN brute force resistance at the **platform level**:

1. Go to **Platform > Global Settings**.
2. Go to the **Break-in Resistance** section and enable or disable **PIN brute force resistance** as appropriate.
PIN brute force resistance is enabled by default.
3. Set the **Maximum PIN failures** allowed before the VMR is blocked (the default is 20).

If you have a lot of meetings with many participants, and with long, complex PINs, you may want to increase the maximum PIN failures limit to tolerate many users mistyping the PIN — however this comes at the expense of reducing resistance to real PIN brute-force attempts.

You can override this setting on a **per location** basis. To do this:

1. Go to **Platform > Locations** and select the required location.
2. Configure **Enable PIN brute force resistance in this location** as appropriate. The options are:
 - **Use Global PIN brute force resistance setting**: as per the global configuration setting.
 - **No**: PIN brute force resistance is disabled for nodes in this location.
 - **Yes**: PIN brute force resistance is enabled for nodes in this location.

When some locations have protection enabled, and other locations do not, the PIN brute force resistance setting is applied according to the location of the node that receives the call signaling.

Default: **Use Global PIN brute force resistance setting**.

VOIP scanner resistance

When VOIP scanner resistance is enabled, Pexip Infinity will temporarily block service access attempts from any unknown source IP address that dials a significant number of incorrect aliases in a short period (and thus may perhaps be attempting to scan your deployment to discover valid aliases to allow the attacker to make improper use of VMRs or gateway rules — such as toll fraud attempts). It blocks **all** new call service access attempts from an IP address if more than 20 incorrect aliases are dialed from that IP

address over SIP, H.323 or WebRTC (Infinity Connect) in a 10 minute window (you can configure the number of allowed incorrect attempts, but you cannot change the time window). There is also a [corresponding alarm](#) raised on the Management Node.

Note that this provides a measure of resistance against scanners such as sipvicious which are sometimes used during toll-fraud attempts, but it does not defend against a determined and patient — or lucky — attacker. Also, enabling this feature could potentially allow a malicious attacker or a legitimate user with incorrect access details to prevent legitimate access to VMRs or other call services for a period, if for example, those users are behind the same firewall as other legitimate users.

To configure VOIP scanner resistance at the **platform level**:

1. Go to **Platform > Global Settings**.
2. Go to the **Break-in Resistance** section and enable or disable **VOIP scanner resistance** as appropriate.
VOIP scanner resistance is enabled by default.
3. Set the **Maximum scanner attempts** i.e. the number of incorrect dial attempts, before the source IP address is blocked (the default is 20).

You can override this setting on a **per location** basis. To do this:

1. Go to **Platform > Locations** and select the required location.
2. Configure **Enable VOIP scanner resistance in this location** as appropriate. The options are:
 - **Use Global VOIP scanner resistance setting**: as per the global configuration setting.
 - **No**: VOIP scanner resistance is disabled for nodes in this location.
 - **Yes**: VOIP scanner resistance is enabled for nodes in this location.

When some locations have protection enabled, and other locations do not, the VOIP scanner resistance setting is applied according to the location of the node that receives the call signaling.

Default: *Use Global VOIP scanner resistance setting*.

Configuring the allow list of IP addresses

You can configure a set of IP addresses that are excluded from the break-in resistance checks. An address can be safelisted for scan attempts and/or incorrect PINs.

- Typically you may want to specify any addresses on a trusted network, where you do not want to penalize genuine mistakes from trusted users.
- You may want to add the address of your reverse proxy (which may itself also employ fail2ban to protect your network at the perimeter) to ignore scan attempts and to ignore incorrect PINs.
- When allowing the address of an upstream SIP or H.323 call control/SBC system, you may want to consider disabling **Ignore incorrect PINs** (i.e. so that Pexip Infinity still performs break-in checks for incorrect PINs from this address) as the call control system cannot police or rate limit incorrect PIN attempts from attackers attempting to brute force a PIN. However, if the call control system has its own VOIP scan resistance behavior — and you trust it — you can enable **Ignore selected scan attempts**.
- For performance reasons, we recommend that you don't add more than a few thousand entries to the allow list table.

To define the allowed addresses:

1. Go to **Call Control > Break-In Attempt Allow List**.
2. Select **Add Allow list address**.
3. Configure the allowed address:

Name	The name of this allow list entry.
Description	A description of the allow list entry.
Network address	The IPv4 or IPv6 address for this allow list IP address range.
Network prefix	The prefix length to use in conjunction with the network address. For example, use a Network address of 10.0.0.0 and a Network prefix of 8 to specify all addresses in the range 10.0.0.0 to 10.255.255.255. You must specify a prefix.

Ignore selected scan attempts	Select this option to allow unlimited scan attempts (incorrect aliases dialled) that are received from addresses in this range. This should only be enabled in trusted environments. Default: not selected.
Ignore incorrect PINs	Select this option to allow unlimited incorrect PIN attempts that are received from addresses in this range. This should only be enabled in trusted environments. Default: not selected.
Entry type	The type of address. This determines whether Pexip Infinity trusts or ignores any security headers (such as X-Forwarded-For headers) from that source. The options are: <ul style="list-style-type: none"> ◦ Proxy: use this for a trusted reverse proxy IP address (or address range). Pexip Infinity trusts security headers from that source to truthfully reflect the IP address of callers/attackers. ◦ User: use this for an end-user IP address range. Pexip Infinity ignores any security headers. Default: User

4. Select **Save**.
5. If required, you can repeat this process to add more addresses.

Break-in prevention policy example log messages

The following examples show messages that may be logged in the **administrator.conference** module of the administrator log (**History & Logs > Administrator Log**) by the break-in prevention policies.

Logged when PIN brute force resistance has temporarily disabled a service, and for all subsequent attempts while the service is blocked:

```
Message="Break-in prevention policy blocking all attempts to join this service." ConferenceAlias="alice" Service="Alice's VMR" Participant="Crooky McCrookface" Protocol="API" Direction="in" Remote-address="10.44.21.35" Reason="Service appears to be under PIN break-in attack" remaining_block_duration_seconds="525"
```

Logged when VOIP scanner resistance has temporarily blocked an address:

```
Message="Participant has been quarantined by Break-in prevention policy due to excessive failed join attempts." Participant="Crooky McCrookface" Protocol="API" Direction="in" Remote-address="10.44.21.35" Reason="Too many attempts to join non-existent aliases" remaining_block_duration_seconds="488"
```

and then any subsequent attempts generate messages such as:

```
Message="Break-in prevention policy rejecting call attempt from quarantined caller." Protocol="API" Direction="in" Local-alias="[u'alice']" Remote-address="10.47.250.169" Reason="Suspicious join attempt rejected" remaining_block_duration_seconds="519"
```

Enabling and disabling chat messages

Conference participants who use a chat-enabled client can send messages and share links with each other within a Virtual Meeting Room or Virtual Auditorium, and when calling each other directly via the Infinity Gateway. Supported clients include Microsoft Skype for Business and Lync clients and Pexip's own Infinity Connect suite.

Chat support is configurable on a platform-wide and per-conference basis, and is enabled by default.

To configure chat at the platform level:

1. Go to **Platform > Global Settings**.
2. From within the **Connectivity** section, deselect or select **Enable chat**.

You can also override the global setting on a per conference basis if required. To do this:

1. Go to **Services > Virtual Meeting Rooms**, **Services > Virtual Auditoriums** or **Services > Scheduled Conferences**.
2. From within the **Advanced Options** section, choose one of the **Enable chat** options:
 - **Use global chat setting**: as per the [global configuration setting](#).
 - **Yes**: chat is enabled.
 - **No**: chat is disabled.

Default: **Use global chat setting**.

When chat is disabled, Infinity Connect clients do not show the chat window.

Providing chat to participants using unsupported clients

Conference participants who are not using one of the supported clients are not able to read or send chat messages.

However, if they have access to a web browser they can use the Infinity Connect web app to join the conference without video or audio. This will give them access to the chat room, plus the ability to view and share presentations, view the participant list, and (if they are Host participants) control aspects of the conference.

For more information on using and administering the Infinity Connect suite of clients, see [Introduction to Infinity Connect](#).



Conferencing Node configuration

Deploying new Conferencing Nodes

Conferencing Nodes are virtual machines that provide the capacity for conferences. They handle all conference media and signaling.

You can deploy your Pexip Infinity platform as either a mix of Proxying Edge Nodes and Transcoding Conferencing Nodes, or as a system that only contains Transcoding Conferencing Nodes.

A typical deployment scenario is to use Proxying Edge Nodes as a front for many privately-addressed Transcoding Conferencing Nodes. Those outward-facing proxying nodes would receive all the signaling and media from endpoints and other external systems, and then forward that media onto other internally-located transcoding nodes to perform the standard Pexip Infinity transcoding, gatewaying and conferencing hosting functions.

There is no limit on the number of Conferencing Nodes that you can add to the Pexip Infinity platform. However, each Conferencing Node must have a unique:

- name
- IP address
- hostname
- DNS name (hostname and domain)


For secure deployments, you should also:

- specify a [SIP TLS FQDN](#)
- replace the self-signed TLS certificates with your own [TLS certificates](#)

Conferencing Nodes can be deployed with dual network interfaces (NICs). Note that you must specify both interface addresses when initially deploying the Conferencing Node; you may also need to assign a static route while deploying the node. For more information, see [Conferencing Nodes with dual network interfaces \(NICs\)](#).

After deploying a new Conferencing Node, it takes approximately 5 minutes before the node is available for conference hosting and for its status to be updated on the Management Node. Until it becomes available, the Management Node reports the status of the Conferencing Node as having a last contacted and last updated date of "Never". "Connectivity lost between nodes" alarms relating to that node may also appear temporarily.

All Conferencing Nodes have identical service configuration, which is obtained automatically from the Management Node.

-  Do not use VMware, Hyper-V or any other tools to clone instances of existing Conferencing Node virtual machines (VMs). Conferencing Nodes must always be created using the Pexip Infinity Administrator interface or management API.

Prerequisites

- All host servers **must** be synchronized with accurate time before you install the Pexip Infinity Management Node or Conferencing Nodes on them.
- You must [enable NTP](#) on the Pexip Infinity Management Node before you deploy any Conferencing Nodes.

Deployment environments

When deploying a new Conferencing Node, a generic instance of a Conferencing Node VM is created, and then it is configured with its specific details such as its IP address and hostname.

This deployment process has been partly automated for on-premises environments using VMware ESXi, Microsoft Hyper-V, KVM or Xen hypervisors. When deploying in a cloud environment such as Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP) or Oracle Cloud Infrastructure, you must also create a suitable VM instance in that environment to host your Conferencing Node before applying a generic configuration file. You can also use a generic VM template to deploy a Conferencing Node in other environments using non-supported hypervisors or orchestration layers.

The different deployment environment options are described in the table below. Your Pexip Infinity platform can contain Conferencing Nodes deployed in any combination of these environments.

Deployment type	Description
Manual (ESXi 6.7 and above)	Pexip Infinity generates an .ova file that you must then deploy from within VMware on to an ESXi host in order to create the Conferencing Node VM.
Manual (ESXi 6.0 and 6.5)	Choose the appropriate ESXi host version for your environment for deployments that use VMware. For more information, see Deploying a Conferencing Node on an ESXi host .
Manual (Hyper-V)	Pexip Infinity generates a file that you must then deploy on a host server running either Microsoft Hyper-V Server 2012 and higher, or Windows Server 2012 and higher, in order to create the Conferencing Node VM. Choose this option for all deployments that use Hyper-V. For more information, see Deploying a Conferencing Node .
Manual (KVM)	Choose this option to generate an .ova file that is suitable for deploying on a KVM host in order to create the Conferencing Node VM. For more information, see Deploying a Conferencing Node on a KVM host .
Manual (Xen)	Choose this option to generate an .ova file that is suitable for deploying on a Xen host in order to create the Conferencing Node VM. For more information, see Deploying a Conferencing Node on a Xen host .
Generic (configuration-only)	This option is most typically used when deploying a Conferencing Node in a cloud environment. For specific deployment information about each platform, see: <ul style="list-style-type: none"> • Deploying Pexip Infinity on Amazon Web Services • Deploying Pexip Infinity on Microsoft Azure • Deploying Pexip Infinity on Google Cloud Platform • Deploying Pexip Infinity on Oracle Cloud Infrastructure <p>Pexip Infinity generates a file containing the configuration of the Conferencing Node. You then upload this file to a generic Conferencing Node that has been created from the Pexip-supplied VM template, in order to configure it with the appropriate settings.</p> <p>You can also choose this option for any deployments that do not use ESXi, Hyper-V, KVM or Xen hypervisors, or for Hyper-V in a cloud-based environment. For more information, see Deploying a Conferencing Node using a generic VM template and configuration file.</p>

Deploying a Conferencing Node on an ESXi host


This process generates an **.ova** file that then must be deployed from within VMware on to an ESXi host.

Note that:

- This file is specific to the Conferencing Node being deployed. It cannot be used to deploy multiple Conferencing Nodes.
- The file is single-use. It cannot be used to re-deploy the same Conferencing Node at a later date. To re-deploy the Conferencing Node, you must first delete it from the Pexip Infinity Management Node and from VMware, and then deploy a new Conferencing Node with the same configuration as the deleted node.
- Before you start, ensure that you are currently using the same machine that you will subsequently use to upload the generated file on to your host server.

To deploy a new Conferencing Node on to a VMware ESXi host:

1. Go to **Platform > Conferencing Nodes** and select **Add Conferencing Node**.
2. You are now asked to provide the network configuration to be applied to the Conferencing Node, by completing the following fields:

Option	Description
Name	Enter the name to use when referring to this Conferencing Node in the Pexip Infinity Administrator interface.
Description	An optional field where you can provide more information about the Conferencing Node.
Role	<p>This determines the Conferencing Node's role:</p> <ul style="list-style-type: none"> ◦ Proxying Edge Node: a Proxying Edge Node handles all media and signaling connections with an endpoint or external device, but does not host any conferences — instead it forwards the media on to a Transcoding Conferencing Node for processing. ◦ Transcoding Conferencing Node: a Transcoding Conferencing Node handles all the media processing, protocol interworking, mixing and so on that is required in hosting Pexip Infinity calls and conferences. When combined with Proxying Edge Nodes, a transcoding node typically only processes the media forwarded on to it by those proxying nodes and has no direct connection with endpoints or external devices. However, a transcoding node can still receive and process the signaling and media directly from an endpoint or external device if required.
Hostname Domain	<p>Enter the hostname and domain to assign to this Conferencing Node. Each Conferencing Node and Management Node must have a unique hostname.</p> <p>The Hostname and Domain together make up the Conferencing Node's DNS name or FQDN. We recommend that you assign valid DNS names to all your Conferencing Nodes.</p>
IPv4 address	Enter the IP address to assign to this Conferencing Node when it is created.
Network mask	Enter the IP network mask to assign to this Conferencing Node.
Gateway IPv4 address	Enter the IP address of the default gateway to assign to this Conferencing Node.
Secondary interface IPv4 address	The optional secondary interface IPv4 address for this Conferencing Node. If configured, this interface is used for signaling and media communications to clients, and the primary interface is used for communication with the Management Node and other Conferencing Nodes.
Secondary interface network mask	The optional secondary interface network mask for this Conferencing Node.
System location	<p>Select the physical location of this Conferencing Node. A system location should not contain a mixture of proxying nodes and transcoding nodes.</p> <p>If the system location does not already exist, you can create a new one here by clicking  to the right of the field. This will open up a new window showing the Add System Location page.</p>
SIP TLS FQDN	A unique identity for this Conferencing Node, used in signaling SIP TLS Contact addresses.
TLS certificate	The TLS certificate to use on this node. This must be a certificate that contains the above SIP TLS FQDN . Each certificate is shown in the format <subject name> (<issuer>) .
IPv6 address	The IPv6 address for this Conferencing Node. Each Conferencing Node must have a unique IPv6 address.
Gateway IPv6 address	<p>The IPv6 address of the default gateway.</p> <p>If this is left blank, the Conferencing Node listens for IPv6 Router Advertisements to obtain a gateway address.</p>
IPv4 static NAT address	The public IPv4 address used by this Conferencing Node when it is located behind a NAT device. Note that if you are using NAT, you must also configure your NAT device to route the Conferencing Node's IPv4 static NAT address to its IPv4 address.

Option	Description
Static routes	From the list of Available Static routes , select the routes to assign to the node, and then use the right arrow to move the selected routes into the Chosen Static routes list.
Enable distributed database	This should usually be enabled (checked) for all Conferencing Nodes that are expected to be "always on", and disabled (unchecked) for nodes that are expected to only be powered on some of the time (e.g. nodes that are likely to only be operational during peak times).
Enable SSH	<p>Determines whether this node can be accessed over SSH.</p> <p><i>Use Global SSH setting:</i> SSH access to this node is determined by the global Enable SSH setting (Platform > Global Settings > Connectivity > Enable SSH).</p> <p><i>Off:</i> this node cannot be accessed over SSH, regardless of the global Enable SSH setting.</p> <p><i>On:</i> this node can be accessed over SSH, regardless of the global Enable SSH setting.</p> <p>Default: <i>Use Global SSH setting</i>.</p>

3. Select **Save**.
4. You are now asked to provide information regarding the deployment environment, CPUs and memory of the Conferencing Node, by completing the following fields:

Option	Description
Deployment type	Select <i>Manual (ESXi 6.7 and above)</i> or <i>Manual (ESXi 6.0 and 6.5)</i> as appropriate.
Number of virtual CPUs to assign	Enter the number of virtual CPUs to assign to the Conferencing Node. We recommend no more than one virtual CPU per physical core, unless you are making use of CPUs that support hyperthreading.
System memory (in megabytes) to assign	<p>Enter the amount of RAM (in megabytes) to assign to the Conferencing Node. The number entered must be a multiple of 4.</p> <p>We recommend 1024 MB (1 GB) RAM for each virtual CPU. The field automatically defaults to the recommended amount, based on the number of virtual CPUs you have entered.</p>
SSH password	<p>Enter the password to use when logging in to this Conferencing Node's Linux operating system over SSH. The username is always <i>admin</i>.</p> <p>Logging in to the operating system is required when changing passwords or for diagnostic purposes only, and should generally be done under the guidance of your Pexip authorized support representative. In particular, do not change any configuration using SSH — all changes should be made using the Pexip Infinity Administrator interface.</p>

5. Select **Download**.
A message appears at the top of the page:
The Conferencing Node image will download shortly or click on the following link (download image now)
After a short while, a file with the name **pexip-*<hostname>*.*<domain>*.ova** is generated and downloaded.
Note that the generated file is only available for your current session so you should download it immediately.
6. When you want to deploy the Conferencing Node VM, use a vSphere client to log in to vCenter Server and select **File > Deploy OVF Template....** Follow the on-screen prompts to deploy the **.ova** file; this is similar to the steps you used when deploying the Management Node. You should always deploy the nodes with Thick Provisioned disks.
After deploying a new Conferencing Node, it takes approximately 5 minutes before the node is available for conference hosting and for its status to be updated on the Management Node. Until it becomes available, the Management Node reports the status of the Conferencing Node as having a last contacted and last updated date of "Never". "Connectivity lost between nodes" alarms relating to that node may also appear temporarily.

Enabling automatic startup

After deploying a new Conferencing Node from VMware, you must enable automatic startup of that virtual machine (VM). In VMware, automatic startup is disabled by default for every new VM — which means that if the host server is powered down for any reason, when it restarts the VM will not restart and must be started manually.

You can only enable automatic startup after the Conferencing Node has been deployed.

To enable automatic startup using the vSphere web client:

1. Log in to the VM manager (vCenter Server).
2. From the navigation panel, select the **Hosts And Clusters** tab and navigate to the host server on which the node's VM is installed.
3. From the main panel, select the **Configure** tab.
4. From the left-hand panel, select **Virtual Machines > VM Startup/Shutdown**.
5. At the top right of the page, select **Edit**.
6. In the **System influence** section, select **Automatically start and stop the virtual machines with the system**.
7. Select **OK**.

Disabling EVC

We strongly recommend that you disable EVC (Enhanced vMotion Compatibility) for any ESXi clusters hosting Conferencing Nodes that include a mix of old and new CPUs. If EVC is enabled on such clusters, the Pexip Infinity platform will run more slowly because the Conferencing Nodes assume they are running on older hardware.

To disable EVC:

1. From the vSphere client's navigation panel, select the cluster.
2. From the main panel, select the **Configure** tab.
3. From the left-hand panel, select **Configuration > VMware EVC**.
The current EVC settings are shown.
4. At the top right of the page, select **Edit**.
5. Select **Disable EVC**.

Deploying a Conferencing Node

This process generates a configuration file that then must be deployed from within Hyper-V.


Note that:

- This file is specific to the Conferencing Node being deployed. It cannot be used to deploy multiple Conferencing Nodes.
- The file is single-use. It cannot be used to re-deploy the same Conferencing Node at a later date. To re-deploy the Conferencing Node, you must first delete it from the Pexip Infinity Management Node and from VMware, and then deploy a new Conferencing Node with the same configuration as the deleted node.
- Before you start, ensure that you are currently using the same machine that you will subsequently use to upload the generated file on to your host server.

To deploy a new Conferencing Node on to a Hyper-V host:

1. Go to **Platform > Conferencing Nodes** and select **Add Conferencing Node**.
2. You are now asked to provide the network configuration to be applied to the Conferencing Node, by completing the following fields:

Option	Description
Name	Enter the name to use when referring to this Conferencing Node in the Pexip Infinity Administrator interface.
Description	An optional field where you can provide more information about the Conferencing Node.

Option	Description
Role	<p>This determines the Conferencing Node's role:</p> <ul style="list-style-type: none"> Proxying Edge Node: a Proxying Edge Node handles all media and signaling connections with an endpoint or external device, but does not host any conferences — instead it forwards the media on to a Transcoding Conferencing Node for processing. Transcoding Conferencing Node: a Transcoding Conferencing Node handles all the media processing, protocol interworking, mixing and so on that is required in hosting Pexip Infinity calls and conferences. When combined with Proxying Edge Nodes, a transcoding node typically only processes the media forwarded on to it by those proxying nodes and has no direct connection with endpoints or external devices. However, a transcoding node can still receive and process the signaling and media directly from an endpoint or external device if required.
Hostname Domain	<p>Enter the hostname and domain to assign to this Conferencing Node. Each Conferencing Node and Management Node must have a unique hostname.</p> <p>The Hostname and Domain together make up the Conferencing Node's DNS name or FQDN. We recommend that you assign valid DNS names to all your Conferencing Nodes.</p>
IPv4 address	Enter the IP address to assign to this Conferencing Node when it is created.
Network mask	Enter the IP network mask to assign to this Conferencing Node.
Gateway IPv4 address	Enter the IP address of the default gateway to assign to this Conferencing Node.
Secondary interface IPv4 address	The optional secondary interface IPv4 address for this Conferencing Node. If configured, this interface is used for signaling and media communications to clients, and the primary interface is used for communication with the Management Node and other Conferencing Nodes.
Secondary interface network mask	The optional secondary interface network mask for this Conferencing Node.
System location	<p>Select the physical location of this Conferencing Node. A system location should not contain a mixture of proxying nodes and transcoding nodes.</p> <p>If the system location does not already exist, you can create a new one here by clicking  to the right of the field. This will open up a new window showing the Add System Location page.</p>
SIP TLS FQDN	A unique identity for this Conferencing Node, used in signaling SIP TLS Contact addresses.
TLS certificate	The TLS certificate to use on this node. This must be a certificate that contains the above SIP TLS FQDN . Each certificate is shown in the format <subject name> (<issuer>) .
IPv6 address	The IPv6 address for this Conferencing Node. Each Conferencing Node must have a unique IPv6 address.
Gateway IPv6 address	<p>The IPv6 address of the default gateway.</p> <p>If this is left blank, the Conferencing Node listens for IPv6 Router Advertisements to obtain a gateway address.</p>
IPv4 static NAT address	The public IPv4 address used by this Conferencing Node when it is located behind a NAT device. Note that if you are using NAT, you must also configure your NAT device to route the Conferencing Node's IPv4 static NAT address to its IPv4 address .
Static routes	From the list of Available Static routes , select the routes to assign to the node, and then use the right arrow to move the selected routes into the Chosen Static routes list.
Enable distributed database	This should usually be enabled (checked) for all Conferencing Nodes that are expected to be "always on", and disabled (unchecked) for nodes that are expected to only be powered on some of the time (e.g. nodes that are likely to only be operational during peak times).

Option	Description
Enable SSH	<p>Determines whether this node can be accessed over SSH.</p> <p><i>Use Global SSH setting:</i> SSH access to this node is determined by the global Enable SSH setting (Platform > Global Settings > Connectivity > Enable SSH).</p> <p><i>Off:</i> this node cannot be accessed over SSH, regardless of the global Enable SSH setting.</p> <p><i>On:</i> this node can be accessed over SSH, regardless of the global Enable SSH setting.</p> <p>Default: <i>Use Global SSH setting.</i></p>

3. Select **Save**.

4. You are now asked to provide information regarding the deployment environment, CPUs and memory of the Conferencing Node, by completing the following fields:

Option	Description
Deployment type	Select <i>Manual (Hyper-V)</i> .
Number of virtual CPUs to assign	Enter the number of virtual CPUs to assign to the Conferencing Node. We recommend no more than one virtual CPU per physical core, unless you are making use of CPUs that support hyperthreading.
System memory (in megabytes) to assign	<p>Enter the amount of RAM (in megabytes) to assign to the Conferencing Node. The number entered must be a multiple of 4.</p> <p>We recommend 1024 MB (1 GB) RAM for each virtual CPU. The field automatically defaults to the recommended amount, based on the number of virtual CPUs you have entered.</p>
SSH password	<p>Enter the password to use when logging in to this Conferencing Node's Linux operating system over SSH. The username is always <i>admin</i>.</p> <p>Logging in to the operating system is required when changing passwords or for diagnostic purposes only, and should generally be done under the guidance of your Pexip authorized support representative. In particular, do not change any configuration using SSH — all changes should be made using the Pexip Infinity Administrator interface.</p>

5. Select **Download**.

A message appears at the top of the page:

The Conferencing Node image will download shortly or click on the following link (download image now)

After a short while, a zip file with the name **pexip-*<hostname>*.*<domain>*.zip** is generated and downloaded.

Note that the generated file is only available for your current session so you should download it immediately.

6. When you wish to deploy the Conferencing Nodes:

- Copy the zip file to the server running Hyper-V and unzip it.
You will see a subfolder called **Virtual Machines**, containing an XML file which contains the configuration for the Conferencing Node VM.
- Open the Hyper-V Manager and select **Import Virtual Machine...**
- Follow the on-screen prompts to deploy the Conferencing Node VM.
When prompted, select the **Virtual Machines** folder and the Hyper-V manager will automatically discover the XML file.
Select the type of import most appropriate for your environment (if you are unsure, select **Restore the virtual machine**).

After deploying a new Conferencing Node, it takes approximately 5 minutes before the node is available for conference hosting and for its status to be updated on the Management Node. Until it becomes available, the Management Node reports the status of the Conferencing Node as having a last contacted and last updated date of "Never". "Connectivity lost between nodes" alarms relating to that node may also appear temporarily.

To look at the steps taken in setting up an example lab deployment of a Conferencing Node in Microsoft Hyper-V, see <https://www.graham-walsh.com/setting-up-pexip-infinity-with-microsoft-hyper-v-server-2012/>.

Enabling automatic startup

By default, virtual machines deployed using Hyper-V are configured to restart automatically if they were running when the host server was shut down or powered off. We recommend that you leave this setting as is for the Management Node and all Conferencing Nodes.

Disabling processor compatibility mode

We strongly recommend that you disable processor compatibility mode on all Hyper-V Conferencing Node virtual machines. If processor compatibility mode is enabled, the Conferencing Node may assume it is running on older hardware, and may stop working, with the message **CPU instruction set is not supported; system will be placed in maintenance mode**.

For more information, see <https://technet.microsoft.com/en-us/library/dn859550.aspx>.

Deploying a Conferencing Node on a KVM host

To deploy a new Conferencing Node onto a KVM host, you must:

1. Use the Pexip Infinity Administrator interface to [generate](#) and download the .ova image.
2. [Convert](#) the .ova image for use with KVM.
3. Create a new volume on your KVM server and [upload](#) the disk image.
4. Create the Conferencing Node [virtual machine](#).
Note that we use the libvirt command line tools to perform the import as they provide greater control than Virtual Machine Manager.
5. Enable the virtual machine for [automatic startup](#).

These steps are described in detail below.

Note that:


- This file is specific to the Conferencing Node being deployed. It cannot be used to deploy multiple Conferencing Nodes.
- The file is single-use. It cannot be used to re-deploy the same Conferencing Node at a later date. To re-deploy the Conferencing Node, you must first delete it from the Pexip Infinity Management Node and from VMware, and then deploy a new Conferencing Node with the same configuration as the deleted node.
- Before you start, ensure that you are currently using the same machine that you will subsequently use to upload the generated file on to your host server.

Generate and download the .ova image

Use the Pexip Infinity Administrator interface to generate and download the .ova image:

1. Go to **Platform > Conferencing Nodes** and select **Add Conferencing Node**.
2. You are now asked to provide the network configuration to be applied to the Conferencing Node, by completing the following fields:

Option	Description
Name	Enter the name to use when referring to this Conferencing Node in the Pexip Infinity Administrator interface.
Description	An optional field where you can provide more information about the Conferencing Node.

Option	Description
Role	<p>This determines the Conferencing Node's role:</p> <ul style="list-style-type: none"> ◦ Proxying Edge Node: a Proxying Edge Node handles all media and signaling connections with an endpoint or external device, but does not host any conferences — instead it forwards the media on to a Transcoding Conferencing Node for processing. ◦ Transcoding Conferencing Node: a Transcoding Conferencing Node handles all the media processing, protocol interworking, mixing and so on that is required in hosting Pexip Infinity calls and conferences. When combined with Proxying Edge Nodes, a transcoding node typically only processes the media forwarded on to it by those proxying nodes and has no direct connection with endpoints or external devices. However, a transcoding node can still receive and process the signaling and media directly from an endpoint or external device if required.
Hostname Domain	<p>Enter the hostname and domain to assign to this Conferencing Node. Each Conferencing Node and Management Node must have a unique hostname.</p> <p>The Hostname and Domain together make up the Conferencing Node's DNS name or FQDN. We recommend that you assign valid DNS names to all your Conferencing Nodes.</p>
IPv4 address	Enter the IP address to assign to this Conferencing Node when it is created.
Network mask	Enter the IP network mask to assign to this Conferencing Node.
Gateway IPv4 address	Enter the IP address of the default gateway to assign to this Conferencing Node.
Secondary interface IPv4 address	The optional secondary interface IPv4 address for this Conferencing Node. If configured, this interface is used for signaling and media communications to clients, and the primary interface is used for communication with the Management Node and other Conferencing Nodes.
Secondary interface network mask	The optional secondary interface network mask for this Conferencing Node.
System location	<p>Select the physical location of this Conferencing Node. A system location should not contain a mixture of proxying nodes and transcoding nodes.</p> <p>If the system location does not already exist, you can create a new one here by clicking  to the right of the field. This will open up a new window showing the Add System Location page.</p>
SIP TLS FQDN	A unique identity for this Conferencing Node, used in signaling SIP TLS Contact addresses.
TLS certificate	The TLS certificate to use on this node. This must be a certificate that contains the above SIP TLS FQDN . Each certificate is shown in the format <subject name> (<issuer>).
IPv6 address	The IPv6 address for this Conferencing Node. Each Conferencing Node must have a unique IPv6 address.
Gateway IPv6 address	<p>The IPv6 address of the default gateway.</p> <p>If this is left blank, the Conferencing Node listens for IPv6 Router Advertisements to obtain a gateway address.</p>
IPv4 static NAT address	The public IPv4 address used by this Conferencing Node when it is located behind a NAT device. Note that if you are using NAT, you must also configure your NAT device to route the Conferencing Node's IPv4 static NAT address to its IPv4 address .
Static routes	From the list of Available Static routes , select the routes to assign to the node, and then use the right arrow to move the selected routes into the Chosen Static routes list.
Enable distributed database	This should usually be enabled (checked) for all Conferencing Nodes that are expected to be "always on", and disabled (unchecked) for nodes that are expected to only be powered on some of the time (e.g. nodes that are likely to only be operational during peak times).

Option	Description
Enable SSH	<p>Determines whether this node can be accessed over SSH.</p> <p><i>Use Global SSH setting:</i> SSH access to this node is determined by the global Enable SSH setting (Platform > Global Settings > Connectivity > Enable SSH).</p> <p><i>Off:</i> this node cannot be accessed over SSH, regardless of the global Enable SSH setting.</p> <p><i>On:</i> this node can be accessed over SSH, regardless of the global Enable SSH setting.</p> <p>Default: <i>Use Global SSH setting.</i></p>

3. Select **Save**.
4. You are now asked to provide information regarding the deployment environment, CPUs and memory of the Conferencing Node, by completing the following fields:

Option	Description
Deployment type	Choose the appropriate hypervisor for your environment.
Number of virtual CPUs to assign	Enter the number of virtual CPUs to assign to the Conferencing Node. We recommend no more than one virtual CPU per physical core, unless you are making use of CPUs that support hyperthreading.
System memory (in megabytes) to assign	<p>Enter the amount of RAM (in megabytes) to assign to the Conferencing Node. The number entered must be a multiple of 4.</p> <p>We recommend 1024 MB (1 GB) RAM for each virtual CPU. The field automatically defaults to the recommended amount, based on the number of virtual CPUs you have entered.</p>
SSH password	<p>Enter the password to use when logging in to this Conferencing Node's Linux operating system over SSH. The username is always <i>admin</i>.</p> <p>Logging in to the operating system is required when changing passwords or for diagnostic purposes only, and should generally be done under the guidance of your Pexip authorized support representative. In particular, do not change any configuration using SSH — all changes should be made using the Pexip Infinity Administrator interface.</p>

5. Select **Download**.

A message appears at the top of the page:

The Conferencing Node image will download shortly or click on the following link (download image now)

After a short while, a zip file with the name **pexip-*<hostname>*.*<domain>*.ova** is generated and downloaded.

Note that the generated file is only available for your current session so you should download it immediately.

Convert the .ova image for use with KVM

To use the Conferencing Node OVA image file with KVM, you must convert it from VMDK to raw format:

1. Copy the downloaded OVA file (named **pexip-*<hostname>*.*<domain>*.ova**) to the server running KVM.
2. Unpack the .ova image, using the command:

```
tar xf pexip-<hostname>.<domain>.ova
```

This unpacks a set of files including **pexip-disk01.vmdk**.

3. Convert the disk image from VMDK to raw, using the command:

```
qemu-img convert -O raw pexip-disk01.vmdk pexip-disk01.raw
```

(This conversion process can take several seconds.)

Create a new volume and upload the disk image

Next, you create a new volume on your KVM server and upload the converted disk image. From within your KVM environment:

1. Use **virsh** to create a new volume on your KVM server:

```
virsh --connect qemu:///<hostname>/system vol-create-as <poolname> <volume_name> 49G --format raw
```

where:

`<hostname>` is the hostname of your KVM server. Note that you can omit the `<hostname>` if you are running **virsh** commands on the local server i.e. you can use `virsh --connect qemu:///system`.

`<poolname>` is the name of the storage pool in which to create the volume; typically you would use **default**. (To determine the storage pools available on the target system, use `virsh --connect qemu:///<hostname>/system pool-list`.)

`<volume_name>` is the name of your new volume.

49G is the virtual size of the volume; always use 49G for a Conferencing Node.

For example:

```
virsh --connect qemu://host1.example.com/system vol-create-as default pexip-conf-01 49G --format raw
```

This example creates a volume named **pexip-conf-01** of size 49 GB and format raw in the storage pool named **default**.

2. Upload the converted disk image to the newly created volume:

```
virsh --connect qemu:///<hostname>/system vol-upload <volume_name> pexip-disk01.raw --pool <poolname>
```

For example:

```
virsh --connect qemu://host1.example.com/system vol-upload pexip-conf-01 pexip-disk01.raw --pool default
```

This example uploads the **pexip-disk01.raw** image to the newly created volume, **pexip-conf-01**, in the storage pool named **default**.

Create the virtual machine

After the disk image has been uploaded, you can create the virtual machine to use it.

Note that we use the libvirt command line tools to perform the import as they provide greater control than Virtual Machine Manager.

1. Identify the filesystem path of the newly uploaded disk image:

```
virsh --connect qemu:///<hostname>/system vol-path <volume_name> --pool <poolname>
```

For example:

```
virsh --connect qemu://host1.example.com/system vol-path pexip-conf-01 --pool default
```

This prints out the absolute path to the disk image file, for example:

```
/var/lib/libvirt/images/pexip-conf-01
```

This path is used in the **disk path** parameter in the next step.

2. Use the **virt-install** command line tool to create the virtual machine:

```
virt-install \
  --import \
  --hvm \
  --name=<vm_name> \
  --arch=x86_64 \
  --vcpus=4 \
  --ram=4096 \
  --cpu host \
  --os-type=linux \
  --connect=qemu:///<hostname>/system \
  --virt-type kvm \
  --disk path=<image_file_path>,bus=virtio,format=raw,cache=none,io=native \
  --network bridge=br0,model=virtio \
  --memballoon virtio \
  --graphics vnc,listen=0.0.0.0,password=<password>
```

This creates a new VM (KVM domain) from the converted disk image.

The command options are described below (items in **bold** may be changed as necessary):

Option	Description
<code>--import</code>	Build guest domain around pre-installed disk image; do not attempt to install a new OS.
<code>--hvm</code>	Create a fully virtualized (i.e. not paravirtualized) VM.
<code>--name=<vm_name></code>	Name of the new VM, where <code><vm_name></code> is, for example, pexip-conf01-vm .
<code>--arch=x86_64</code>	CPU architecture of new VM (must be x84_64).
<code>--vcpus=4</code>	Number of CPUs allocated to new VM. By default, this is 4 for the Conferencing Node.
<code>--ram=4096</code>	Memory allocated to new VM (in megabytes).
<code>--cpu host</code>	Expose all host CPU capabilities to new VM (CPUID).

Option	Description
<code>--os-type=linux</code>	The guest OS is Linux.
<code>--connect=qemu://<hostname>/system</code>	Connect to KVM on the target system, where <hostname> is the hostname of your KVM server.
<code>--virt-type kvm</code>	Use KVM to host the new VM.
<code>--disk path=<image_file_path>, bus=virtio, format=raw, cache=none, io=native</code>	<ul style="list-style-type: none"> Define the location of the disk image file, where <image_file_path> is as determined in the previous step, for example <code>/var/lib/libvirt/images/pexip-conf-01</code>. Expose it to the guest on the virtio paravirtualized bus (as opposed to IDE/SCSI). Define the image file as being in raw format. Instruct the host system not to cache the disk contents in memory. Use the native IO backend to access the disk device.
<code>--network bridge=br0,model=virtio</code>	<ul style="list-style-type: none"> Create a network interface connected to the br0 bridge interface on the host. Expose it to the guest as a virtio paravirtualized NIC.
<code>--memballoon virtio</code>	Expose the virtio memory balloon to the guest.
<code>--graphics vnc,listen=0.0.0.0,password=<password></code>	Expose the graphical console over VNC, listening on 0.0.0.0 (i.e. all addresses on the target system) and with an access password of <password> .

You may receive a warning "Unable to connect to graphical console: virt-viewer not installed"; if so, this message can be safely ignored.

After the VM has been created, it may be managed using the Virtual Machine Manager desktop interface (virt-manager application) or via the command line interface (virsh).

The new node should start automatically. If it does not you can use the Virtual Machine Manager to start the node, or the CLI command:

```
virsh --connect qemu://<hostname>/system start <vm_name>
```

Note that you can list existing VMs by using `virsh --connect qemu://<hostname>/system list`


After deploying a new Conferencing Node, it takes approximately 5 minutes before the node is available for conference hosting and for its status to be updated on the Management Node. Until it becomes available, the Management Node reports the status of the Conferencing Node as having a last contacted and last updated date of "Never". "Connectivity lost between nodes" alarms relating to that node may also appear temporarily.

Enabling automatic startup

After deploying a new Conferencing Node in KVM, you should enable automatic startup of that virtual machine (VM). In KVM, automatic startup is disabled by default for every new VM. This means that if the host server is powered down for any reason, when it restarts the VM will not restart and must be started manually.

You can only enable automatic startup after the Conferencing Node has been deployed.

To enable automatic startup using Virtual Machine Manager:

1. Connect to the Virtual Machine Manager (virt-manager) that is managing the node's VM.
2. Select the node's VM and then, from the toolbar, select the **Show the virtual machine console and details** icon . A new window for that VM is opened.
3. If necessary, select **View > Details** to display the VM information.
4. From the sidebar menu, select **Boot Options**.
5. Select the **Start virtual machine on host boot up** check box.
6. Select **Apply**.

Deploying a Conferencing Node on a Xen host

To deploy a new Conferencing Node onto a Xen host, you must:

1. Use the Pexip Infinity Administrator interface to [generate](#) and download the .ova image.
2. [Convert](#) the .ova image for use with Xen.
3. Create a new volume on your Xen server and [upload](#) the disk image.
4. Create the Conferencing Node [virtual machine](#).
Note that we use the libvirt command line tools to perform the import as they provide greater control than Virtual Machine Manager.
5. Enable the virtual machine for [automatic startup](#).

These steps are described in detail below.

Note that:


- This file is specific to the Conferencing Node being deployed. It cannot be used to deploy multiple Conferencing Nodes.
- The file is single-use. It cannot be used to re-deploy the same Conferencing Node at a later date. To re-deploy the Conferencing Node, you must first delete it from the Pexip Infinity Management Node and from VMware, and then deploy a new Conferencing Node with the same configuration as the deleted node.
- Before you start, ensure that you are currently using the same machine that you will subsequently use to upload the generated file on to your host server.

Generate and download the .ova image

Use the Pexip Infinity Administrator interface to generate and download the .ova image:

1. Go to **Platform > Conferencing Nodes** and select **Add Conferencing Node**.
2. You are now asked to provide the network configuration to be applied to the Conferencing Node, by completing the following fields:

Option	Description
Name	Enter the name to use when referring to this Conferencing Node in the Pexip Infinity Administrator interface.
Description	An optional field where you can provide more information about the Conferencing Node.
Role	<p>This determines the Conferencing Node's role:</p> <ul style="list-style-type: none"> ◦ Proxying Edge Node: a Proxying Edge Node handles all media and signaling connections with an endpoint or external device, but does not host any conferences — instead it forwards the media on to a Transcoding Conferencing Node for processing. ◦ Transcoding Conferencing Node: a Transcoding Conferencing Node handles all the media processing, protocol interworking, mixing and so on that is required in hosting Pexip Infinity calls and conferences. When combined with Proxying Edge Nodes, a transcoding node typically only processes the media forwarded on to it by those proxying nodes and has no direct connection with endpoints or external devices. However, a transcoding node can still receive and process the signaling and media directly from an endpoint or external device if required.
Hostname Domain	<p>Enter the hostname and domain to assign to this Conferencing Node. Each Conferencing Node and Management Node must have a unique hostname.</p> <p>The Hostname and Domain together make up the Conferencing Node's DNS name or FQDN. We recommend that you assign valid DNS names to all your Conferencing Nodes.</p>
IPv4 address	Enter the IP address to assign to this Conferencing Node when it is created.
Network mask	Enter the IP network mask to assign to this Conferencing Node.
Gateway IPv4 address	Enter the IP address of the default gateway to assign to this Conferencing Node.

Option	Description
Secondary interface IPv4 address	The optional secondary interface IPv4 address for this Conferencing Node. If configured, this interface is used for signaling and media communications to clients, and the primary interface is used for communication with the Management Node and other Conferencing Nodes.
Secondary interface network mask	The optional secondary interface network mask for this Conferencing Node.
System location	<p>Select the physical location of this Conferencing Node. A system location should not contain a mixture of proxying nodes and transcoding nodes.</p> <p>If the system location does not already exist, you can create a new one here by clicking  to the right of the field. This will open up a new window showing the Add System Location page.</p>
SIP TLS FQDN	A unique identity for this Conferencing Node, used in signaling SIP TLS Contact addresses.
TLS certificate	The TLS certificate to use on this node. This must be a certificate that contains the above SIP TLS FQDN . Each certificate is shown in the format <subject name> (<issuer>).
IPv6 address	The IPv6 address for this Conferencing Node. Each Conferencing Node must have a unique IPv6 address.
Gateway IPv6 address	<p>The IPv6 address of the default gateway.</p> <p>If this is left blank, the Conferencing Node listens for IPv6 Router Advertisements to obtain a gateway address.</p>
IPv4 static NAT address	The public IPv4 address used by this Conferencing Node when it is located behind a NAT device. Note that if you are using NAT, you must also configure your NAT device to route the Conferencing Node's IPv4 static NAT address to its IPv4 address .
Static routes	From the list of Available Static routes , select the routes to assign to the node, and then use the right arrow to move the selected routes into the Chosen Static routes list.
Enable distributed database	This should usually be enabled (checked) for all Conferencing Nodes that are expected to be "always on", and disabled (unchecked) for nodes that are expected to only be powered on some of the time (e.g. nodes that are likely to only be operational during peak times).
Enable SSH	<p>Determines whether this node can be accessed over SSH.</p> <p>Use Global SSH setting: SSH access to this node is determined by the global Enable SSH setting (Platform > Global Settings > Connectivity > Enable SSH).</p> <p>Off: this node cannot be accessed over SSH, regardless of the global Enable SSH setting.</p> <p>On: this node can be accessed over SSH, regardless of the global Enable SSH setting.</p> <p>Default: Use Global SSH setting.</p>

- Select **Save**.
- You are now asked to provide information regarding the deployment environment, CPUs and memory of the Conferencing Node, by completing the following fields:

Option	Description
Deployment type	Choose the appropriate hypervisor for your environment.
Number of virtual CPUs to assign	Enter the number of virtual CPUs to assign to the Conferencing Node. We recommend no more than one virtual CPU per physical core, unless you are making use of CPUs that support hyperthreading.

Option	Description
System memory (in megabytes) to assign	<p>Enter the amount of RAM (in megabytes) to assign to the Conferencing Node. The number entered must be a multiple of 4.</p> <p>We recommend 1024 MB (1 GB) RAM for each virtual CPU. The field automatically defaults to the recommended amount, based on the number of virtual CPUs you have entered.</p>
SSH password	<p>Enter the password to use when logging in to this Conferencing Node's Linux operating system over SSH. The username is always admin.</p> <p>Logging in to the operating system is required when changing passwords or for diagnostic purposes only, and should generally be done under the guidance of your Pexip authorized support representative. In particular, do not change any configuration using SSH — all changes should be made using the Pexip Infinity Administrator interface.</p>

5. Select Download.

A message appears at the top of the page:

The Conferencing Node image will download shortly or click on the following link (download image now)

After a short while, a zip file with the name **pexip-<hostname>.<domain>.ova** is generated and downloaded.

Note that the generated file is only available for your current session so you should download it immediately.

Convert the .ova image for use with Xen

To use the Conferencing Node OVA image file with Xen, you must convert it from VMDK to raw format:

1. Copy the downloaded OVA file (named **pexip-<hostname>.<domain>.ova**) to the server running Xen.
2. Unpack the .ova image, using the command:

```
tar xf pexip-<hostname>.<domain>.ova
```

This unpacks a set of files including **pexip-disk01.vmdk**.

3. Convert the disk image from VMDK to raw, using the command:

```
qemu-img convert -O raw pexip-disk01.vmdk pexip-disk01.raw
```

(This conversion process can take several seconds.)

Create a new volume and upload the disk image

Next, you create a new volume on your Xen server and upload the converted disk image. From within your Xen environment:

1. Use **virsh** to create a new volume on your Xen server:

```
virsh --connect xen://<hostname>/ vol-create-as <poolname> <volume_name> 49G --format raw
```

where:

<hostname> is the hostname of your Xen server. Note that you can omit the **<hostname>** if you are running **virsh** commands on the local server i.e. you can use **virsh --connect xen:///**.

<poolname> is the name of the storage pool in which to create the volume; typically you would use **default**. (To determine the storage pools available on the target system, use **virsh --connect xen://<hostname>/ pool-list**.)

<volume_name> is the name of your new volume.

49G is the virtual size of the volume; always use 49G for a Conferencing Node.

For example:

```
virsh --connect xen://host1.example.com/ vol-create-as default pexip-conf-01 49G --format raw
```

This example creates a volume named **pexip-conf-01** of size 49 GB and format raw in the storage pool named **default**.

2. Upload the converted disk image to the newly created volume:

```
virsh --connect xen://<hostname>/ vol-upload <volume_name> pexip-disk01.raw --pool <poolname>
```

For example:

```
virsh --connect xen://host1.example.com/ vol-upload pexip-conf-01 pexip-disk01.raw --pool default
```

This example uploads the **pexip-disk01.raw** image to the newly created volume, **pexip-conf-01**, in the storage pool named **default**.

Create the virtual machine

After the disk image has been uploaded, you can create the virtual machine to use it.

Note that we use the libvirt command line tools to perform the import as they provide greater control than Virtual Machine Manager.

1. Identify the filesystem path of the newly uploaded disk image:

```
virsh --connect xen://<hostname>/ vol-path <volume_name> --pool <poolname>
```

For example:

```
virsh --connect xen://host1.example.com/ vol-path pexip-conf-01 --pool default
```

This prints out the absolute path to the disk image file, for example:

```
/var/lib/libvirt/images/pexip-conf-01
```

This path is used in the **disk path** parameter in the next step.

2. Use the **virt-install** command line tool to create the virtual machine:

```
virt-install \
--import \
--hvm \
--name=<vm_name> \
--arch=x86_64 \
--vcpus=4 \
--ram=4096 \
--os-type=linux \
--connect=xen://<hostname>/ \
--virt-type xen \
--disk path=<image_file_path>,bus=xen,format=raw,driver_name=qemu,cache=none,io=native \
--network bridge=xenbr0,model=e1000 \
--memballoon xen \
--graphics vnc,listen=0.0.0.0,password=<password>
```

This creates a new VM (Xen domain) from the converted disk image.

The command options are described below (items in **bold** may be changed as necessary):

Option	Description
--import	Build guest domain around pre-installed disk image; do not attempt to install a new OS.
--hvm	Create a fully virtualized (i.e. not paravirtualized) VM.
--name=<vm_name>	Name of the new VM, where <vm_name> is, for example, pexip-conf01-vm .
--arch=x86_64	CPU architecture of new VM (must be x84_64).
--vcpus=4	Number of CPUs allocated to new VM. By default, this is 4 for the Conferencing Node.
--ram=4096	Memory allocated to new VM (in megabytes).
--os-type=linux	The guest OS is Linux.
--connect=xen://<hostname>/	Connect to Xen on the target system, where <hostname> is the hostname of your Xen server.
--virt-type xen	Use Xen to host the new VM.
--disk path=<image_file_path>,bus=xen,format=raw,driver_name=qemu,cache=none,io=native	<ul style="list-style-type: none"> Define the location of the disk image file, where <image_file_path> is as determined in the previous step, for example /var/lib/libvirt/images/pexip-conf-01. Expose it to the guest on the xen paravirtualized bus (as opposed to IDE/SCSI). Define the image file as being in raw format and the driver as qemu. Instruct the host system not to cache the disk contents in memory. Use the native IO backend to access the disk device.
--network bridge=xenbr0,model=e1000	<ul style="list-style-type: none"> Create a network interface connected to the xenbr0 bridge interface on the host. Expose it to the guest as an e1000 NIC.
--memballoon xen	Expose the xen memory balloon to the guest.

Option	Description
<code>--graphics</code> <code>vnc,listen=0.0.0.0,</code> <code>password=<password></code>	Expose the graphical console over VNC, listening on 0.0.0.0 (i.e. all addresses on the target system) and with an access password of <password> .

You may receive a warning "Unable to connect to graphical console: virt-viewer not installed"; if so, this message can be safely ignored.

After the VM has been created, it may be managed using the Virtual Machine Manager desktop interface (virt-manager application) or via the command line interface (virsh).

The new node should start automatically. If it does not you can use the Virtual Machine Manager to start the node, or the CLI command:

```
virsh --connect xen://<hostname>/ start <vm_name>
```

Note that you can list existing VMs by using `virsh --connect xen://<hostname>/ list`


After deploying a new Conferencing Node, it takes approximately 5 minutes before the node is available for conference hosting and for its status to be updated on the Management Node. Until it becomes available, the Management Node reports the status of the Conferencing Node as having a last contacted and last updated date of "Never". "Connectivity lost between nodes" alarms relating to that node may also appear temporarily.

Enabling automatic startup

After deploying a new Conferencing Node in Xen, you should enable automatic startup of that virtual machine (VM). In Xen, automatic startup is disabled by default for every new VM. This means that if the host server is powered down for any reason, when it restarts the VM will not restart and must be started manually.

You can only enable automatic startup after the Conferencing Node has been deployed.

To enable automatic startup using Virtual Machine Manager:

1. Connect to the Virtual Machine Manager (virt-manager) that is managing the node's VM.
2. Select the node's VM and then, from the toolbar, select the **Show the virtual machine console and details** icon . A new window for that VM is opened.
3. If necessary, select **View > Details** to display the VM information.
4. From the sidebar menu, select **Boot Options**.
5. Select the **Start virtual machine on host boot up** check box.
6. Select **Apply**.

Deploying a Conferencing Node using a generic VM template and configuration file

You can also use a generic Pexip Infinity VM template to deploy Conferencing Nodes on other non-supported hypervisors or orchestration layers.

Creating a new generic Conferencing Node is a two-step process:

1. Creating a generic instance of a Conferencing Node VM, using a template provided by Pexip.
2. Configuring the VM with the details of the specific Conferencing Node being deployed, using a file generated from the Pexip Infinity Management Node.

Prerequisites

- The Conferencing Node must be deployed in a VM environment that supports address assignment by DHCP.
- You must know the IP address that will initially be assigned to the Conferencing Node. You will use this IP address to connect to the VM in order to upload the configuration file, but this configuration file may then assign a new IP address to the Conferencing Node.

Creating a generic Conferencing Node VM

To create a new generic instance of a Conferencing Node using the VM template:

1. Within your chosen environment, go to <https://dl.pexip.com/infinity/index.html>, select the appropriate directory for your software version, and then download one of the following files and use it to create a generic instance of a Conferencing Node:
 - **Pexip_Infinity_v26_generic_ConfNode_<build>.ova** in environments that take .ova or .ovf files as input.
 - **Pexip_Infinity_v26_HyperV_ConfNode_<build>.zip** for Hyper-V in cloud-based environments or other orchestration layers where standard deployment is problematic.


Ensure you are using a VM template with the same Pexip Infinity software version as that which is currently running on the Management Node. This includes dot releases — so for example, for a v25.1 Management Node you must install a v25.1 Conferencing Node rather than a v25 Conferencing Node. If the Management Node has been upgraded, you will need to download the Conferencing Node VM template corresponding to that software version. For more information, see [Upgrading configuration-only deployments](#).

2. Within your hypervisor, configure the generic Conferencing Node VM with the appropriate number of virtual CPUs and amount of RAM.

Generating and downloading the configuration file

1. From the Pexip Infinity Administrator interface, go to **Platform > Conferencing Nodes** and select **Add Conferencing Node**.
2. You are now asked to provide the network configuration to be applied to the Conferencing Node, by completing the following fields:

Option	Description
Name	Enter the name to use when referring to this Conferencing Node in the Pexip Infinity Administrator interface.
Description	An optional field where you can provide more information about the Conferencing Node.
Role	<p>This determines the Conferencing Node's role:</p> <ul style="list-style-type: none"> ◦ Proxying Edge Node: a Proxying Edge Node handles all media and signaling connections with an endpoint or external device, but does not host any conferences — instead it forwards the media on to a Transcoding Conferencing Node for processing. ◦ Transcoding Conferencing Node: a Transcoding Conferencing Node handles all the media processing, protocol interworking, mixing and so on that is required in hosting Pexip Infinity calls and conferences. When combined with Proxying Edge Nodes, a transcoding node typically only processes the media forwarded on to it by those proxying nodes and has no direct connection with endpoints or external devices. However, a transcoding node can still receive and process the signaling and media directly from an endpoint or external device if required.
Hostname Domain	<p>Enter the hostname and domain to assign to this Conferencing Node. Each Conferencing Node and Management Node must have a unique hostname.</p> <p>The Hostname and Domain together make up the Conferencing Node's DNS name or FQDN. We recommend that you assign valid DNS names to all your Conferencing Nodes.</p>
IPv4 address	Enter the IP address to assign to this Conferencing Node when it is created.
Network mask	Enter the IP network mask to assign to this Conferencing Node.
Gateway IPv4 address	Enter the IP address of the default gateway to assign to this Conferencing Node.
Secondary interface IPv4 address	The optional secondary interface IPv4 address for this Conferencing Node. If configured, this interface is used for signaling and media communications to clients, and the primary interface is used for communication with the Management Node and other Conferencing Nodes.
Secondary interface network mask	The optional secondary interface network mask for this Conferencing Node.

Option	Description
System location	<p>Select the physical location of this Conferencing Node. A system location should not contain a mixture of proxying nodes and transcoding nodes.</p> <p>If the system location does not already exist, you can create a new one here by clicking  to the right of the field. This will open up a new window showing the Add System Location page.</p>
SIP TLS FQDN	A unique identity for this Conferencing Node, used in signaling SIP TLS Contact addresses.
TLS certificate	The TLS certificate to use on this node. This must be a certificate that contains the above SIP TLS FQDN . Each certificate is shown in the format <subject name> (<issuer>) .
IPv6 address	The IPv6 address for this Conferencing Node. Each Conferencing Node must have a unique IPv6 address.
Gateway IPv6 address	<p>The IPv6 address of the default gateway.</p> <p>If this is left blank, the Conferencing Node listens for IPv6 Router Advertisements to obtain a gateway address.</p>
IPv4 static NAT address	The public IPv4 address used by this Conferencing Node when it is located behind a NAT device. Note that if you are using NAT, you must also configure your NAT device to route the Conferencing Node's IPv4 static NAT address to its IPv4 address .
Static routes	From the list of Available Static routes , select the routes to assign to the node, and then use the right arrow to move the selected routes into the Chosen Static routes list.
Enable distributed database	This should usually be enabled (checked) for all Conferencing Nodes that are expected to be "always on", and disabled (unchecked) for nodes that are expected to only be powered on some of the time (e.g. nodes that are likely to only be operational during peak times).
Enable SSH	<p>Determines whether this node can be accessed over SSH.</p> <p>Use Global SSH setting: SSH access to this node is determined by the global Enable SSH setting (Platform > Global Settings > Connectivity > Enable SSH).</p> <p>Off: this node cannot be accessed over SSH, regardless of the global Enable SSH setting.</p> <p>On: this node can be accessed over SSH, regardless of the global Enable SSH setting.</p> <p>Default: Use Global SSH setting.</p>

- Select **Save**.
- You are now asked to provide information regarding the deployment environment, CPUs and memory of the Conferencing Node, by completing the following fields:

Option	Description
Deployment type	Select Generic (configuration-only) .
Number of virtual CPUs to assign	You can ignore this field (leave it as default) for Generic (configuration-only) deployments.
System memory (in megabytes) to assign	You can ignore this field (leave it as default) for Generic (configuration-only) deployments.
SSH password	<p>Enter the password to use when logging in to this Conferencing Node's Linux operating system over SSH. The username is always admin.</p> <p>Logging in to the operating system is required when changing passwords or for diagnostic purposes only, and should generally be done under the guidance of your Pexip authorized support representative. In particular, do not change any configuration using SSH — all changes should be made using the Pexip Infinity Administrator interface.</p>

- Select **Download**.

A message appears at the top of the page:

The Conferencing Node image will download shortly or click on the following link ([download image now](#))

After a short while, a zip file with the name **pexip-<hostname>.<domain>.xml** is downloaded.

Note that the generated file is only available for your current session so you should download it immediately.

6. Browse to **<https://<conferencing-node-ip>:8443/>** and use the form provided to upload the configuration file to the Conferencing Node VM.

The Conferencing Node will apply the configuration and then reboot. When it has rebooted, it will connect to the Management Node in the usual way.

After deploying a new Conferencing Node, it takes approximately 5 minutes before the node is available for conference hosting and for its status to be updated on the Management Node. Until it becomes available, the Management Node reports the status of the Conferencing Node as having a last contacted and last updated date of "Never". "Connectivity lost between nodes" alarms relating to that node may also appear temporarily.

Enabling automatic startup

We recommend that all virtual machines are configured to restart automatically if they were running when the host server was shut down or powered off.

Assigning hostnames and FQDNs

The Management Node and each Conferencing Node within your deployment must have a unique hostname.

Hostnames can be made of a combination of numbers, letters, and the hyphen (-) character, and are case-insensitive. Hostnames cannot be entirely numeric.

A Conferencing Node's assigned hostname/FQDN is not used in any call signaling, and can be different from the actual DNS FQDN you might use to point to that Conferencing Node.

The node's [SIP TLS FQDN](#) is used:

- in SIP signaling (if configured)
- MS-SIP signaling (mandatory)
- for any self-referential redirects in web requests.

However, we recommend that the hostname and the domain combination that you assign should match the DNS name or FQDN that you will use to refer to the node for call routing. This makes it easier to manage your deployment, for example by:

- allowing you to use more manageable [TLS certificates](#)
- making the system easier to access for the purposes of troubleshooting.

To assign a DNS name to a Conferencing Node, enter a valid **Hostname** and **Domain** combination when first [deploying the Conferencing Node](#).



Configuring existing Conferencing Nodes

To reconfigure or delete an existing Conferencing Node, go to **Platform > Conferencing Nodes** and click on the name of the Conferencing Node.

Any changes to the configuration of the Conferencing Node should be made using the Pexip Infinity Administrator interface. Do not make any changes using other tools (such as VMware or Hyper-V); doing so may cause the Conferencing Node to fail. The only exception is any change to the CPU and RAM allocated to the Conferencing Node — these changes should be made using the hypervisor and should only be done while the Conferencing Node is powered off.

When reconfiguring existing Conferencing Nodes, the following information can be changed:

Option	Description
Name	The name used to refer to this Conferencing Node in the Pexip Infinity Administrator interface. Each Conferencing Node must have a unique name.

Option	Description
Description	An optional field where you can provide more information about the Conferencing Node.
Role	<p>This determines the Conferencing Node's role:</p> <ul style="list-style-type: none"> • Proxying Edge Node: a Proxying Edge Node handles all media and signaling connections with an endpoint or external device, but does not host any conferences — instead it forwards the media on to a Transcoding Conferencing Node for processing. • Transcoding Conferencing Node: a Transcoding Conferencing Node handles all the media processing, protocol interworking, mixing and so on that is required in hosting Pexip Infinity calls and conferences. When combined with Proxying Edge Nodes, a transcoding node typically only processes the media forwarded on to it by those proxying nodes and has no direct connection with endpoints or external devices. However, a transcoding node can still receive and process the signaling and media directly from an endpoint or external device if required. <p> If you change the role of a Conferencing Node, all existing calls will be disconnected and the Conferencing Node will be restarted.</p>
System location	<p>The physical location of this Conferencing Node. A system location should not contain a mixture of proxying nodes and transcoding nodes.</p> <p> If you change the system location of a Conferencing Node, all existing calls will be disconnected and the Conferencing Node will be restarted.</p>
Enable maintenance mode	<p>While maintenance mode is enabled, this Conferencing Node will not accept any new conference instances. For more information, see Taking a Conferencing Node out of service.</p> <p>The maintenance mode setting will persist after a reboot.</p>
SIP TLS FQDN	A unique identity for this Conferencing Node, used in signaling SIP TLS Contact addresses.
TLS certificate	The TLS certificate to use on this node. This must be a certificate that contains the above SIP TLS FQDN. Each certificate is shown in the format <subject name> (<issuer>).
IPv6 address	The IPv6 address for this Conferencing Node. Each Conferencing Node must have a unique IPv6 address.
Gateway IPv6 address	<p>The IPv6 address of the default gateway.</p> <p>If this is left blank, the Conferencing Node listens for IPv6 Router Advertisements to obtain a gateway address.</p>
IPv4 static NAT address	The public IPv4 address used by this Conferencing Node when it is located behind a NAT device.
Static routes	From the list of Available Static routes , select the routes to assign to the node, and then use the right arrow to move the selected routes into the Chosen Static routes list.
Enable SSH	<p>Determines whether this node can be accessed over SSH.</p> <p>Use Global SSH setting: SSH access to this node is determined by the global Enable SSH setting (Platform > Global Settings > Connectivity > Enable SSH).</p> <p>Off: this node cannot be accessed over SSH, regardless of the global Enable SSH setting.</p> <p>On: this node can be accessed over SSH, regardless of the global Enable SSH setting.</p> <p>Default: Use Global SSH setting.</p>

You can also change the node's SNMP settings (see [Monitoring via SNMP](#) for more information):

Option	Description
SNMP mode	<p>Configures the SNMP access mode for the selected node:</p> <p><i>Off</i>: SNMP is disabled. You cannot use SNMP to query the node for its status.</p> <p><i>SNMPv2c read-only</i>: enables insecure, read-only access.</p> <p><i>SNMPv3 read-only</i>: enables secure, read-only access, using the authPriv security level with SHA1 authentication and AES 128-bit encryption.</p> <p>When enabled, access is provided to the basic RFC 1213 MIB-II tree (1.3.6.1.2.1).</p> <p>Default: <i>Off</i>.</p>
SNMP community	<p>The SNMP group to which this node belongs. This setting applies to SNMPv2c only.</p> <p>Default: public</p>
SNMPv3 username	The node's SNMPv3 username, used to authenticate SNMPv3 requests.
SNMPv3 privacy password	<p>The node's SNMPv3 privacy password used for encrypting messages between the node and the management station.</p> <p>AES encryption must be used; DES is not supported.</p>
SNMPv3 authentication password	<p>The node's SNMPv3 authentication password, used to authenticate the associated username.</p> <p>The SHA authentication protocol must be used; MD5 is not supported.</p>
SNMP system contact	The contact details (for example, email address) of the person responsible for this particular node.
SNMP system location	A description of the node's location.

Other details of the Conferencing Node that cannot be directly changed (however, see [Changing the node's IP address](#) below) are also shown on this page for your information, as follows:

Option	Description
IPv4 address	The IPv4 address of this Conferencing Node.
Network mask	The IP network mask for this Conferencing Node.
Gateway IPv4 address	The IPv4 address of the default gateway.
Secondary interface IPv4 address and network mask	<p>The optional secondary interface IPv4 address and network mask for this Conferencing Node.</p> <p>These fields are only displayed if the Conferencing Node has been deployed with dual network interfaces.</p>
Hostname Domain	<p>The DNS hostname and domain of this Conferencing Node. Together these make up the machine's FQDN, or <i>DNS Name</i> in VMware.</p> <p>For more information, see Assigning hostnames and FQDNs.</p>
Enable distributed database	<p>This should usually be enabled (checked) for all Conferencing Nodes that are expected to be "always on", and disabled (unchecked) for nodes that are expected to only be powered on some of the time (e.g. nodes that are likely to only be operational during peak times).</p> <p>You should avoid frequent toggling of this setting. When changing this setting on multiple Conferencing Nodes, update one node at a time, waiting a few minutes before updating the next node.</p>

SIP TLS FQDN

If your deployment includes Microsoft Skype for Business / Lync, you **must** configure each Conferencing Node with a **SIP TLS FQDN**. However, for security purposes we recommend that all deployments use SIP TLS FQDNs.

In summary, the node's SIP TLS FQDN is used:

- in SIP signaling (if configured)
- MS-SIP signaling (mandatory)
- for any self-referential redirects in web requests.

To configure the **SIP TLS FQDN**, go to **Platform > Conferencing Nodes** and select each Conferencing Node in turn.

The FQDN in the **SIP TLS FQDN** field is used in SIP signaling over TLS, and specifies the identity that the Conferencing Node service will use when identifying itself to the systems connecting to it. Each Conferencing Node must have a unique **SIP TLS FQDN**.

The **SIP TLS FQDN**:

- must match one of the identities returned in the certificate for the Conferencing Node, and
- must have an entry in DNS

so that the identity of the Conferencing Node can be verified by the systems connecting to it.

The **SIP TLS FQDN** can be the same as the Conferencing Node's FQDN (made up of its Hostname and Domain).

If the **SIP TLS FQDN** field is left blank, the IP address of the Conferencing Node is used in SIP TLS signaling, and, depending on your call control configuration, this may result in calls failing.

For more details on the use of domain certificates in SIP, see [section 4 of RFC 5922](#).


Changing the node's IP address

You can change the IPv4 address of a Conferencing Node by adding a new node with the new address, and deleting the node with the old address.

If you want to preserve the Conferencing Node's existing FQDN, you should:

1. Put the Conferencing Node whose IP address you want to change into [maintenance mode](#).
2. When all calls on that Conferencing Node have terminated, [delete the Conferencing Node](#).
3. Wait at least 90 seconds to allow the deletion to be synchronized across the platform.
4. [Add a new Conferencing Node](#) with the new IP address (but using the same **Name**, **Hostname** and **Domain** etc. as used before, if required).
5. If necessary, make any changes to your call control system so that calls are routed to the Conferencing Node's new IP address.

If you want to maintain full service capacity, and do not need to preserve the node's FQDN, then you can add the new node with the new IP address before deleting the old node with the old IP address. There is no limit on the number of Conferencing Nodes that you can add to the Pexip Infinity platform.

-  Do not attempt to change the IP address of a Conferencing Node using utilities available in external tools (such as VMware or Hyper-V) because this will cause Pexip Infinity services to fail.

Deleting Conferencing Nodes

When deleting a Conferencing Node, you must first remove its details from the Pexip Infinity platform using the Pexip Infinity Administrator interface, and then delete it from the hypervisor/cloud platform.

To delete a Conferencing Node:

1. Put the Conferencing Node into [maintenance mode](#) and wait until all calls on it have terminated.
2. From the Pexip Infinity Administrator interface, go to **Platform > Conferencing Nodes**.
3. Select the Conferencing Node to be deleted, and from the **Action** drop-down menu select **Delete selected Conferencing Nodes**.
4. Select **Go** and on the following page confirm that you want to delete the selected Conferencing Node by selecting **Yes, I'm sure**.
5. Shut down and remove the Conferencing Node VM.

For on-premises installations:

- a. Log in to the VM manager, shut down the deleted Conferencing Node and then power it off.
- b. Right-click on the Conferencing Node and select **Delete from Disk** (VMware) or **Delete** (Hyper-V / KVM / Xen).


For cloud-based installations:

- a. Log in to your cloud provider's management portal.
- b. Terminate the Conferencing Node instance.
- c. If using Azure, delete the resource group and storage containers which hold the instance.

Pexip Infinity conference types

About Pexip Infinity conferences

The Pexip Infinity platform offers a variety of conference types and services:

- [Virtual Meeting Rooms](#) and [Virtual Auditoriums](#) are used to hold conferences, share presentations, and chat. Participants can join over audio or video from any location using virtually any type of communications tool, such as Microsoft Skype for Business / Lync, a traditional conferencing endpoint, a mobile telephone, or a Pexip Infinity Connect client.
 - The [Virtual Reception](#) IVR service provides a way for conference participants who cannot dial Virtual Meeting Room and Virtual Auditorium aliases directly, to access these services from a central point using DTMF tones. It can also be used to route calls via the Infinity Gateway.
 - The [Pexip Infinity Distributed Gateway](#):
 - Enables endpoints within your deployment to make direct calls to other endpoints. As with calls into VMRs, the gateway can interwork the protocols and media formats used by each type of device (SIP, H.323, WebRTC etc).
 - Enables any type of endpoint, including traditional VTC endpoints, to join externally-hosted meeting services such as Skype for Business / Lync meetings, Microsoft Teams, and Google Meet,
 - Can be used with call control systems and other third party services to enable calls from your deployment to external devices including PSTN and mobile phones.
 - [VMR Scheduling for Exchange](#) integrates Pexip Infinity with Microsoft Exchange. It enables Microsoft Outlook desktop and Web App users (using Office 365, Exchange 2013 or Exchange 2016) to schedule meetings using Pexip VMRs as a meeting resource.
 - Pexip's [One-Touch Join](#) (OTJ) allows users to schedule a meeting in Microsoft Outlook or Google Calendar and include in the invitation a meeting room with a supported Cisco or Poly videoconferencing endpoint, so that the endpoint in the chosen meeting room displays a Join button just before the meeting is scheduled to begin. Participants can then simply walk into the room and select the button, and the endpoint will automatically dial in to the meeting.
 - A [Test Call Service](#) provides a test loopback service that allows users to check the quality of their video and audio (i.e. that their local camera, microphone and speakers are working properly), and verifies that they can connect to a Conferencing Node.
-  The [Live view](#) page (Status > Live View) lets you review current and historic usage charts showing a breakdown of participants by location, protocol, license type and the different conference types being hosted.

About Virtual Meeting Rooms (VMRs) and Virtual Auditoriums

Conferences take place in Virtual Meeting Rooms (VMRs) and Virtual Auditoriums (a type of VMR that is optimized for use by a small number of Hosts and a large number of Guests). They act as a digital space that participants using virtually any type of device can join for reliable, HD group video meetings.

When configuring your VMR and Virtual Auditorium services, you define the [aliases](#) that are used by conference participants to access that service. Your services can be PIN-protected, and you can also assign Host and Guest privileges to conference participants (see [About PINs, Hosts and Guests](#)), and you can [limit the number of participants](#) that can join a specific meeting.

All Virtual Meeting Rooms and Virtual Auditoriums can be accessed via any Conferencing Node. When a Conferencing Node receives a call to a particular Virtual Meeting Room or Virtual Auditorium alias, it creates a conference instance based on that service's settings. In this way, resources are not used until the first caller actually places a call into the conference. When another endpoint places a call to an alias that belongs to the same Virtual Meeting Room or Virtual Auditorium, the call is placed into the existing conference instance.

You can change the audio and video prompts presented to participants when they are accessing these services by applying [themes](#).

VMRs do not consume resources on your deployment unless they are actually being used to host a conference. The number of VMRs that can be in use at the same time is limited only by the size of your Pexip Infinity deployment (in terms of server capacity and call licenses).

VMRs can be [bulk-provisioned](#) from directory information contained in a Windows Active Directory LDAP server, or any other LDAP-accessible database.

- See [Configuring Virtual Meeting Rooms \(VMRs\)](#) for full information about how to configure a VMR.
- See [Configuring Virtual Auditoriums](#) for full information about how to configure a Virtual Auditorium.

Allowing end-users to manage their own VMRs

VMR settings such as Host and Guest PIN numbers and whether to display participant names are initially assigned when the VMR is created by the administrator.

The Pexip VMR self-service portal is a separately-installable component that allows end-users to manage their personal Virtual Meeting Room without having to send requests to their administrator to change the configuration of their VMR.

Virtual Meeting Rooms versus Virtual Auditoriums

A Virtual Auditorium is a meeting space that is optimized for use by a small number of Hosts and a large number of Guests, for example:

- A lecture, where the lecturer joins as the Host and all the students join as Guests
- An internal team meeting, where the department heads join as Hosts and all the managers join as Guests.

Similarities

Virtual Auditoriums share many of the same features as Virtual Meeting Rooms:

- Each Virtual Auditorium has one or more aliases associated with it. Participants access the Virtual Auditorium by dialing any one of its aliases — this will route them all to the same conference. For more information, see [About aliases and access numbers](#).
- They use PINs to protect access, and to differentiate between Host and Guest participants. For more information, see [About PINs, Hosts and Guests](#).
- You can [place a limit on the number of participants](#) that can access a Virtual Auditorium at any one time.
- Participants can access Virtual Auditoriums from any video endpoint, or by using one of Pexip's Infinity Connect clients — for more information see [Introduction to Infinity Connect](#).
- There is no limit on the number of Virtual Auditoriums that can be configured on your Pexip Infinity platform. Virtual Auditoriums do not consume resources on your deployment unless they are actually being used to host a conference. Unless you [manually restrict access](#), the number of participants who can access a particular Virtual Auditorium, and the number of Virtual Auditoriums that can be in use at the same time, are limited only by the size of your Pexip Infinity deployment.

Differences

Virtual Auditoriums differ from Virtual Meeting Rooms in that:

- **Raised hands:** Guest participants using Infinity Connect clients have an option to raise their hand, to indicate that they wish to speak. (Generally in a Virtual Auditorium, the Host will have muted all Guests, but they can unmute individual guests.) Host participants using the Infinity Connect clients can lower guests' hands, as can the Guest themselves. However, there is a `raiseHandInVmr` customization option that can be used to also enable the raise hands feature in Virtual Meeting Rooms.
- Administrators can configure a Virtual Auditorium so that Hosts and Guests [see different layouts](#). In a Virtual Meeting Room, all participants see the same layout.
- In a Virtual Auditorium:
 - **Guest** participants cannot see other Guests — they only see the Host(s)
 - **Host** participants see all other Hosts first in the video thumbnails (in order of which Host spoke most recently), followed by Guests (again, in order of who spoke most recently).

In a Virtual Meeting Room all participants can see all other participants, in order of who spoke most recently, and regardless of whether they are Hosts or Guests.

- In a Virtual Auditorium, Guest participants are not shown the [streaming indicator](#) and so are not aware if the conference is being streamed or recorded. In a Virtual Meeting Room, Guest participants are shown the streaming indicator.
- Administrators can configure a Virtual Auditorium so that when a presentation is being shown, the person showing the presentation [is fixed in the main speaker position](#). In a Virtual Meeting Room, this option is not available; the main speaker position is always voice-switched, showing the current speaker.

Configuring Virtual Meeting Rooms (VMRs)

A Virtual Meeting Room (VMR) is a virtual meeting space that is always available to hold a Pexip Infinity conference. It can host any number of people from any type of device. Each VMR has one or more aliases associated with it, and participants access the

conference by dialing one of these aliases. VMRs can be PIN-protected, and you can also assign Host and Guest privileges to conference participants.

To create a new VMR, go to **Services > Virtual Meeting Rooms** and select **Add Virtual Meeting Room**.


To edit an existing VMR, or to view its details (including all the aliases associated with it), go to **Services > Virtual Meeting Rooms** and select the name of the VMR.


As an alternative to manually configuring your VMRs:

- You can import Virtual Meeting Rooms and device aliases from directory information contained in a Windows Active Directory LDAP server, or any other LDAP-accessible database. For more information, see [Provisioning VMRs, devices and users from Active Directory via LDAP](#).
- You can bulk import basic VMR configuration from a CSV file. See [Bulk import/export of service configuration data](#).
- VMRs are also added by the VMR Scheduling for Exchange service; these are shown on a separate page (**Services > Scheduled Conferences**). For more information, see [VMR Scheduling for Exchange](#).

Any configuration changes made to VMRs are replicated to all Conferencing Nodes within around 90 seconds and applied to any subsequent conferences in that VMR. If there are any conferences already in place that use the VMR, any attempts to join it after the configuration has been replicated may be affected by the new configuration settings. Therefore, we do not recommend changing VMR configuration while a conference is in progress in it, as this may lead to an inconsistent user experience.

When adding or editing Virtual Meeting Rooms, the options are:

Option	Description
Name	<p>The name used to refer to this VMR.</p> <p> If you can access this VMR via a Virtual Reception then the VMR Name entered here is shown to conference participants as they are transferred into the VMR (it is overlaid onto the <code>virtual_reception_connecting</code> splash screen of the theme associated with the Virtual Reception that is transferring the call).</p>
Description	A description of the VMR.
Creation time	This read-only field shows the date and time when this record was first configured.
Host PIN	<p>This optional field allows you to set a secure access code that must be entered by VMR participants before they can join the conference.</p> <p>If Allow Guests is set to Yes, then the Host PIN will apply to the conference Host(s) only.</p> <p>For more information, see About PINs, Hosts and Guests.</p> <ul style="list-style-type: none"> • PINs must use the digits 0-9 only. • PINs may optionally end with #. • PINs must be between 4–20 digits long, including any #.
Allow Guests	<p>Yes: the conference can have two types of participants: Hosts and Guests. You must configure a Host PIN to be used by the Hosts. You can optionally configure a Guest PIN; if you do not configure a Guest PIN, Guests can join without a PIN, but the meeting will not start until the first Host has joined.</p> <p>No: all participants have Host privileges.</p> <p>Default: No.</p>

Option	Description
Guest PIN	<p>This optional field allows you to set a secure access code that must be entered by VMR Guests before they can join the conference.</p> <p>For more information, see About PINs, Hosts and Guests.</p> <ul style="list-style-type: none"> Host PINs and Guest PINs must be different. PINs must use the digits 0-9 only. PINs may optionally end with #. PINs must be between 4–20 digits long, including any #. If the Host PIN ends in # and a Guest PIN is used, the Guest PIN must also end with #. If # is not used, Host PINs and Guest PINs must have the same number of digits. You cannot configure a Guest PIN unless you have already configured a Host PIN.
View	<p>The layout controls the maximum number of other participants that each participant will see, and how the participants are arranged on the screen. For more information, see Selecting the layout seen by participants.</p> <p>Default: <i>Large main speaker and up to 7 other participants (1+7 layout)</i>.</p>
Show name of active speaker	<p>When active speaker display is enabled, the display name or alias of the current speaker is shown across the bottom of their video image. This option is only available in Adaptive Composition layouts.</p> <p>Default: <i>No</i>.</p>
Show names of participants	<p>If participant name overlays are enabled, the display names or aliases of all participants are shown in a text overlay along the bottom of their video image.</p> <p>Default: <i>No</i>.</p>
Theme	<p>The theme for use with this VMR. For more information, see Customizing conference images and voice prompts using themes.</p> <p>Default: <i><use Default theme></i> (the global default theme is used).</p>
Owner's email address	<p>The email address of the owner of the VMR.</p> <ul style="list-style-type: none"> For VMRs created using LDAP synchronization, provisioning messages for this VMR will be sent to this address. For VMRs created by manual input, CSV import, or via the API, this field is optional.
Advanced options	
Automatically Dialed Participants	<p>When a conference begins in this VMR, a call will be placed automatically to any participants selected here. To add an Automatically Dialed Participant that is not already on the list, select the  icon to the right of the selection fields.</p> <p>For more information, see Automatically dialing out to a participant from a conference.</p>
Guests can present	<p>Controls whether the Guests in the conference are allowed to present content.</p> <ul style="list-style-type: none"> <i>Yes</i>: Guests and Hosts can present into the conference <i>No</i>: only Hosts can present <p>Default: <i>Yes</i></p>
Enable chat	<p>Whether chat messaging is enabled for the conference:</p> <ul style="list-style-type: none"> <i>Use global chat setting</i>: as per the global configuration setting. <i>Yes</i>: chat is enabled. <i>No</i>: chat is disabled. <p>Default: <i>Use global chat setting</i>.</p>
Maximum inbound call bandwidth (kbps)	<p>Specifying a maximum inbound call bandwidth will limit the bandwidth of media received by Pexip Infinity from each individual participant dialed in to this VMR. For more information see Managing and restricting call bandwidth.</p>

Option	Description
Maximum outbound call bandwidth (kbps)	Specifying a maximum outbound call bandwidth will limit the bandwidth of media sent from Pexip Infinity to each individual participant dialed in to this VMR. For more information see Managing and restricting call bandwidth .
Conference capabilities	Allows you to limit the media content of the conference. For more information, see Controlling media capability . Default: <i>Main video + presentation</i> .
Maximum call quality	Controls the maximum call quality for participants connecting to this service: <ul style="list-style-type: none"> <i>Use global setting</i>: use the global maximum call quality setting. <i>SD</i>: each participant is limited to SD quality. <i>HD</i>: each participant is limited to HD (720p) quality. <i>Full HD (1080p)</i>: allows any endpoint capable of Full HD to send and receive its main video at 1080p. Default: <i>Use global setting</i>
Media encryption	Controls the media encryption requirements for participants connecting to this service. <ul style="list-style-type: none"> <i>Use global setting</i>: use the global media encryption setting. <i>Best effort</i>: each participant will use media encryption if their device supports it, otherwise the connection will be unencrypted. <i>Required</i>: all participants (including RTMP participants) must use media encryption. <i>No encryption</i>: all H.323, SIP and MS-SIP participants must use unencrypted media. (RTMP participants will use encryption if their device supports it, otherwise the connection will be unencrypted.) Default: <i>Use global setting</i>
Participant limit	This optional field allows you to limit the number of participants allowed to join this VMR. For more information see Limiting the number of participants .
Service tag	This optional field lets you assign a unique identifier to this service, which you can then use to track use of the service .
VMR origin	This read-only field shows the name of the service (if any) used to create this VMR. <ul style="list-style-type: none"> For VMRs created using LDAP synchronization, this will be LDAP Sync template: followed by the name of the LDAP sync template being used. For VMRs created by manual input, CSV import, or via the API, this will be blank.
Aliases	
Alias: #1	
Alias	<p>The alias that, when received by Pexip Infinity, will cause it to route the call to this service.</p> <p>The alias entered here must match the alias as it is received by Pexip Infinity. Wildcards and regular expressions are not supported.</p> <p>In most cases, the alias received by Pexip Infinity will be the same as the alias that the conference participant dialed from their endpoint, but there are some exceptions, described in About aliases and access numbers.</p> <p>You may also want to define multiple aliases for the same service to ensure that it can be accessed by devices and protocols that enforce specific alias formats — for more information, see Using multiple aliases to access the same service.</p>
Description	An optional description of the alias. This is useful if you have more than one alias for a service. Note that this description may be displayed to end users on registered Infinity Connect clients who are performing a directory search.
Add another Alias	Select this option if you want the VMR to be accessible by more than one alias. For more information, see Using multiple aliases to access the same service .

Allowing end-users to manage their own VMRs

VMR settings such as Host and Guest PIN numbers and whether to display participant names are initially assigned when the VMR is created by the administrator.

The Pexip VMR self-service portal is a separately-installable component that allows end-users to manage their personal Virtual Meeting Room without having to send requests to their administrator to change the configuration of their VMR.

Configuring Virtual Auditoriums

Large Pexip Infinity conferences can take place in Virtual Auditoriums. Each Virtual Auditorium has one or more aliases associated with it, and participants access the conference by dialing one of these aliases. Virtual Auditoriums can be PIN-protected, and you can also assign Host and Guest privileges to conference participants.


To create a new Virtual Auditorium, go to **Services > Virtual Auditoriums** and select **Add Virtual Auditorium**.


To edit an existing Virtual Auditorium, or to view its details (including all the aliases associated with it), go to **Services > Virtual Auditoriums** and click on the name of the Virtual Auditorium.

You cannot convert an existing Virtual Meeting Room into a Virtual Auditorium. Instead, you must create a new Virtual Auditorium and reassign the existing aliases to it. For more information, see [Changing from a Virtual Meeting Room to a Virtual Auditorium and vice versa](#).

Any changes you make to Virtual Auditorium configuration will be replicated to all Conferencing Nodes within around 90 seconds and will be applied to any subsequent conferences in that Virtual Auditorium. If there are any conferences already in place that use the Virtual Auditorium, any attempts to join it after the configuration has been replicated may be affected by the new configuration settings. For this reason, we do not recommend changing Virtual Auditorium configuration while a conference is in progress because this will lead to an inconsistent user experience.

When adding or editing Virtual Auditoriums, the options are:

Option	Description
Name	<p>The name used to refer to this Virtual Auditorium.</p> <p> If you can access this Virtual Auditorium via a Virtual Reception then the Virtual Auditorium Name entered here is shown to conference participants as they are transferred into the Virtual Auditorium (it is overlaid onto the <code>virtual_reception_connecting</code> splash screen of the theme associated with the Virtual Reception that is transferring the call).</p>
Description	A description of the Virtual Auditorium.
Creation time	This read-only field shows the date and time when this record was first configured.
Host PIN	<p>This optional field allows you to set a secure access code that must be entered by participants before they can join the conference.</p> <p>If Allow Guests is set to Yes, then the Host PIN will apply to the conference Host(s) only.</p> <p>For more information, see About PINs, Hosts and Guests.</p> <ul style="list-style-type: none"> PINs must use the digits 0-9 only. PINs may optionally end with #. PINs must be between 4–20 digits long, including any #.
Allow Guests	<p>Yes: the conference can have two types of participants: Hosts and Guests. You must configure a Host PIN to be used by the Hosts. You can optionally configure a Guest PIN; if you do not configure a Guest PIN, Guests can join without a PIN, but the meeting will not start until the first Host has joined.</p> <p>No: all participants have Host privileges.</p> <p>Default: <i>No</i>.</p>

Option	Description
Guest PIN	<p>This optional field allows you to set a secure access code that must be entered by Guests before they can join the conference.</p> <p>For more information, see About PINs, Hosts and Guests.</p> <ul style="list-style-type: none"> • Host PINs and Guest PINs must be different. • PINs must use the digits 0-9 only. • PINs may optionally end with #. • PINs must be between 4–20 digits long, including any #. • If the Host PIN ends in # and a Guest PIN is used, the Guest PIN must also end with #. • If # is not used, Host PINs and Guest PINs must have the same number of digits. • You cannot configure a Guest PIN unless you have already configured a Host PIN.
Host view	<p>The maximum number of other participants that Hosts will see, and the layout used to show them. For more information, see Selecting the layout seen by participants.</p> <p>Default: <i>Large main speaker and up to 7 other participants (1+7 layout)</i>.</p>
Guest view	<p>The maximum number of Host participants that each Guest will see, and the layout used to show them. For more information, see Selecting the layout seen by participants.</p> <p>Default: <i>Large Host speaker and up to 7 other Hosts (1+7 layout)</i>.</p>
Show names of participants	<p>If participant name overlays are enabled, the display names or aliases of all participants are shown in a text overlay along the bottom of their video image.</p> <p>Default: <i>No</i>.</p>
Mute all Guests	<p>When enabled, Guest participants will be muted when they first join the conference. After the conference has started, Hosts can use the Infinity Connect client to unmute Guests, either individually or as a group.</p> <p>Default: <i>No</i>.</p>
Lock presenter as main speaker	<p>When a presentation is being shown, whether the main speaker position shows the presenter or the current speaker. For more information, see Selecting the layout seen by participants.</p> <p><i>Yes</i>: When a presentation is being shown, the main speaker position will always show the video image from the endpoint that is showing the presentation, even if others are speaking.</p> <p><i>No</i>: When a presentation is being shown the main speaker position will be voice-switched as usual.</p> <p>Default: <i>No</i>.</p>
Theme	<p>The theme for use with this Virtual Auditorium. For more information, see Customizing conference images and voice prompts using themes.</p> <p>Default: <i><use Default theme></i> (the global default theme is used).</p>
Advanced options	
Automatically dialed participants	<p>When a conference begins in this Virtual Auditorium, a call will be placed automatically to any participants selected here. To add an Automatically Dialed Participant that is not already on the list, select the  icon to the right of the selection fields.</p> <p>For more information, see Automatically dialing out to a participant from a conference.</p>
Guests can present	<p>Controls whether the Guests in the conference are allowed to present content.</p> <ul style="list-style-type: none"> • <i>Yes</i>: Guests and Hosts can present into the conference • <i>No</i>: only Hosts can present <p>Default: <i>Yes</i></p>

Option	Description
Enable chat	<p>Whether chat messaging is enabled for the conference:</p> <ul style="list-style-type: none"> <i>Use global chat setting:</i> as per the global configuration setting. <i>Yes:</i> chat is enabled. <i>No:</i> chat is disabled. <p>Default: <i>Use global chat setting</i>.</p>
Maximum inbound call bandwidth (kbps)	<p>Enter a value in this field to limit the bandwidth of media being received by Pexip Infinity from each individual participant dialed in to this Virtual Auditorium. For more information see Managing and restricting call bandwidth.</p>
Maximum outbound call bandwidth (kbps)	<p>Enter a value in this field to limit the bandwidth of media being sent from Pexip Infinity to each individual participant dialed in to this Virtual Auditorium. For more information see Managing and restricting call bandwidth.</p>
Conference capabilities	<p>Allows you to limit the media content of the conference. For more information, see Controlling media capability.</p> <p>Default: <i>Main video + presentation</i>.</p>
Maximum call quality	<p>Controls the maximum call quality for participants connecting to this service:</p> <ul style="list-style-type: none"> <i>Use global setting:</i> use the global maximum call quality setting. <i>SD:</i> each participant is limited to SD quality. <i>HD:</i> each participant is limited to HD (720p) quality. <i>Full HD (1080p):</i> allows any endpoint capable of Full HD to send and receive its main video at 1080p. <p>Default: <i>Use global setting</i></p>
Media encryption	<p>Controls the media encryption requirements for participants connecting to this service.</p> <ul style="list-style-type: none"> <i>Use global setting:</i> use the global media encryption setting. <i>Best effort:</i> each participant will use media encryption if their device supports it, otherwise the connection will be unencrypted. <i>Required:</i> all participants (including RTMP participants) must use media encryption. <i>No encryption:</i> all H.323, SIP and MS-SIP participants must use unencrypted media. (RTMP participants will use encryption if their device supports it, otherwise the connection will be unencrypted.) <p>Default: <i>Use global setting</i></p>
Participant limit	<p>This optional field allows you to limit the number of participants allowed to join this Virtual Auditorium. For more information see Limiting the number of participants.</p>
Service tag	<p>This optional field lets you assign a unique identifier to this service, which you can then use to track use of the service.</p>
Aliases	
Alias: #1	
Alias	<p>The alias that, when received by Pexip Infinity, will cause it to route the call to this service.</p> <p>The alias entered here must match the alias as it is received by Pexip Infinity. Wildcards and regular expressions are not supported.</p> <p>In most cases, the alias received by Pexip Infinity will be the same as the alias that the conference participant dialed from their endpoint, but there are some exceptions, described in About aliases and access numbers.</p> <p>You may also want to define multiple aliases for the same service to ensure that it can be accessed by devices and protocols that enforce specific alias formats — for more information, see Using multiple aliases to access the same service.</p>

Option	Description
Description	An optional description of the alias. This is useful if you have more than one alias for a service. Note that this description may be displayed to end users on registered Infinity Connect clients who are performing a directory search.
Add another Alias	Select this option if you want the Virtual Auditorium to be accessible by more than one alias. For more information, see Using multiple aliases to access the same service .

Changing from a Virtual Meeting Room to a Virtual Auditorium and vice versa

While it is not possible to change an existing Virtual Meeting Room to a Virtual Auditorium (and vice versa), you can change the service to which an alias is routed. For example, if your sales team already have a Virtual Meeting Room with an alias `meet.sales@example.com` and you want these conferences to take advantage of the features available to a Virtual Auditorium instead, then:

1. Set up a new Virtual Auditorium (**Services > Virtual Auditoriums > Add Virtual Auditorium**) with the appropriate configuration, but do not configure any aliases.
2. Save the new Virtual Auditorium.
3. Go to the **Aliases** page (**Services > Aliases**) and select the alias of the existing Virtual Meeting Room.
4. From the **Service name** drop-down list, select the name of the new Virtual Auditorium.
5. Repeat for all aliases belonging to the existing Virtual Meeting Room.

All calls made to any of the aliases that previously belonged to the Virtual Meeting Room will now be routed to the Virtual Auditorium. The Virtual Meeting Room will still exist, but will not have any aliases, so no calls can be made to it.

About the Virtual Reception IVR service

The Virtual Reception IVR service provides a way for conference participants who cannot dial Virtual Meeting Room and Virtual Auditorium aliases directly, to access these services from a central point using DTMF tones. It can also be used to route calls via the Pexip Distributed Gateway to join an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.

Using the Virtual Reception to access VMRs

Generally, in order to access a Virtual Meeting Room, participants dial one of the Virtual Meeting Room's aliases directly from their endpoint. However, some conference participants may not be able to do this. Reasons might include:

- The participant's endpoint might not support dialing by URI - they can only dial by IP address.
- The participant might be using an audio-only PSTN telephone and can only make calls to direct dial numbers.
- Your enterprise might have a limited number of direct dial numbers that can be used for audio participants, and you do not want to allocate one per VMR.
- Your enterprise uses local, toll-free telephone numbers that audio-only users can dial to access your VMRs and you don't want to have one of these for every VMR in your enterprise.

To allow for such cases, Pexip Infinity enables you to set up a single direct dial number or IP address that participants can dial to access a single, central Virtual Reception. From here they can use the DTMF tones on their endpoint to enter the number of the specific VMR they want to join.

To implement this you must:

1. [Create and configure your Virtual Reception](#).
2. Ensure that every Virtual Meeting Room that you want to be accessible from the Virtual Reception has at least one alias that consists of digits only. For more information, see [Using multiple aliases to access the same service](#).
3. Provide conference participants with the combination of Virtual Reception alias followed by the Virtual Auditorium or Virtual Meeting Room number that they must dial to access the conference.

Providing telephone access to Virtual Meeting Rooms

For information on how to set up audio-only PSTN or mobile telephone access to your Virtual Meeting Rooms and Virtual Auditoriums via a Virtual Reception, see [Integrating with telephone systems \(PSTN\)](#).

Virtual Receptions and Skype for Business / Lync clients

To allow Skype for Business / Lync clients to use Virtual Receptions, you must ensure that the target VMR is configured with at least one alias in the form of a SIP URI that is routable by the SfB/Lync client. (This is in addition to the digits-only VMR alias used by the Virtual Reception.) Pexip Infinity chooses the first alias in the VMR's configuration that is a valid SIP URI; it also replaces the selected alias's domain with the **Pexip Infinity domain (for Lync / Skype for Business integration)** (from the relevant system location or global setting as appropriate, if configured).

Note that the SfB/Lync client will be transferred into the destination VMR or gateway call as an audio-only participant, although they can subsequently escalate to two-way video if required (providing the **Call capability** configuration of the VMR or Call Routing Rule allows it).

Including the numeric alias of the VMR in the Virtual Reception dial string

SIP and H.323 endpoints and Skype for Business / Lync clients can optionally bypass having to enter the destination alias via DTMF tones.

They can do this by including the numeric alias of the VMR in their dial string when dialing the Virtual Reception. The dial string should be in the format: `<reception_alias>*<destination_alias>@<domain>`.

For example, if the alias of the Virtual Reception is `ivr@example.com` and the numeric alias of the target Virtual Meeting Room is `1234`, then the endpoint can dial `ivr**1234@example.com` to be transferred directly into the VMR.

Note that H.323 devices can also use the dial format `<reception_alias>#<destination_alias>@<domain>`.

Restricting or transforming the aliases entered into a Virtual Reception

To increase the security of your Virtual Reception services you may want to:

- restrict the aliases or alias patterns that can be entered into a Virtual Reception
- optionally transform the entered alias before the Virtual Reception attempts to route the call to that new destination.

You can do this when configuring your Virtual Reception by expanding the **Advanced Options** and specifying the **Destination alias regex match** and **Destination alias regex replace string** fields.

For example, you could:

- lock down the number ranges that can be reached via the Virtual Reception, e.g. by specifying a regex match of `8(.*)` to only allow calls to numbers starting with "8"
- restrict and then transform the aliases that are entered, e.g. by specifying a regex match of `(\d{4})` to only allow 4 digit extensions to be entered and then to prepend the entered alias with a prefix of "88" by entering a replace string of `88\1`
- append a domain onto any numbers entered (so that you don't need to configure so many alternative VMR aliases), e.g. by specifying a regex match of `(\d{4})` and then appending a domain by using a replace string of `\1@example.com`
- only allow access to a specified shortlist of aliases, e.g. by configuring a match string of `(1111|1112|1113)` to reject any other alias than 1111, 1112 or 1113.

Using the Virtual Reception with the Infinity Gateway

You can also use the Virtual Reception to route calls via the Infinity Gateway. This would allow you, for example, to route phone calls towards a Cisco VCS as E.164 numbers, or to join an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.

i If your environment includes a PSTN gateway or uses an ITSP (Internet telephony service provider), consider the potential for toll fraud if you have Call Routing Rules that can route calls to the PSTN gateway or ITSP, or if you allow conference participants to dial out to other participants via the PSTN gateway or ITSP. See [PSTN gateways and toll fraud](#) for more information.

Joining scheduled and ad hoc Skype for Business / Lync meetings

See [Configuring Pexip Infinity as a Skype for Business / Lync gateway](#) for more information.

Joining scheduled and ad hoc Microsoft Teams meetings

See [Integrating Microsoft Teams with Pexip Infinity](#) for more information.

Joining scheduled and ad hoc Google Meet meetings

Routing phone calls towards a Cisco VCS as E.164 numbers

To implement this you must:

1. Create and configure your Virtual Reception in the normal way.
2. Configure a Call Routing Rule that matches the desired E.164 pattern and then routes the call via the specified Cisco VCS (where the Cisco VCS is the SIP proxy or H.323 gatekeeper as specified in the rule).
Ensure that the E.164 pattern does not match a Virtual Meeting Room alias (as Virtual Meeting Room routing takes precedence).

Now, if someone dials in to the Virtual Reception and then enters the E.164 number that matches the Call Routing Rule, their call will be routed via the Cisco VCS.

Configuring Virtual Reception IVRs

The Virtual Reception IVR service allows participants to use DTMF tones to select the Virtual Meeting Room they want to join. It provides an alternative means to access VMRs for participants who cannot dial VMR aliases directly.

It can also be used in combination with the Infinity Gateway to:

- route calls to an external call control system, for example, to route phone calls towards a Cisco VCS
- join an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.

To create or edit a Virtual Reception IVR service:

1. Go to **Services > Virtual Receptions**.
2. To create a new Virtual Reception, select **Add Virtual Reception**, or to edit an existing Virtual Reception, click on its name.

The following options are available:

Option	Description
Name	Enter the name you will use to refer to this Virtual Reception.
Creation time	This read-only field shows the date and time when this record was first configured.
Description	An optional field where you can add details about the Virtual Reception.
Theme	Select the theme to apply to this Virtual Reception. For more information, see Customizing conference images and voice prompts using themes . If you leave this blank, the default Pexip theme is used.
Virtual Reception type	The type of Virtual Reception: <ul style="list-style-type: none"> ◦ Regular: the default type of Virtual Reception, used for routing calls to VMRs, or to other devices and call control systems via the Infinity Gateway. ◦ Lync / Skype for Business: a special type of Virtual Reception, used when you want to provide an IVR gateway to scheduled and ad hoc Skype for Business / Lync meetings. ◦ Google Meet: a special type of Virtual Reception, used when you want to provide an IVR gateway to scheduled and ad hoc Google Meet meetings. ◦ Microsoft Teams: a special type of Virtual Reception, used when you want to provide an IVR gateway to scheduled and ad hoc Microsoft Teams meetings. Default: <i>Regular</i>

Lync / Skype for Business options (when a Virtual Reception type of Lync / Skype for Business is selected)

Option	Description
Lync / Skype for Business server	<p>The SfB/Lync server to use to resolve the SfB/Lync Conference ID entered by the user.</p> <p>You must then ensure that your Call Routing Rule that routes calls to your SfB/Lync environment has Match against full alias URI selected and a Destination alias regex match in the style <code>.*@example.com.*</code>.</p> <p>For more information, see Configuring Pexip Infinity as a Skype for Business / Lync gateway.</p>
Lookup location	If specified, a Conferencing Node in this system location will perform the SfB/Lync Conference ID lookup on the SfB/Lync server. If a location is not specified, the transcoding node hosting the Virtual Reception will perform the lookup.
Google Meet options (when a Virtual Reception type of <i>Google Meet</i> is selected)	
Access token	Select the name of the access token to use to resolve Google Meet IDs. When configuring a Virtual Reception it does not matter if you use a trusted or untrusted access token.
Lookup location	Specify the location that contains the Conferencing Nodes (typically Proxying Edge Nodes) that will perform the service lookup (meeting ID verification) on Google Meet.
Post-lookup regex match	<p>An optional regular expression used to match against the meeting code entered by the caller into the Virtual Reception. This is typically used in conjunction with the Post-lookup regex replace string to transform the meeting code into a distinct alias pattern that will match a Call Routing Rule configured to route calls into Google Meet conferences.</p> <p>For example, you would typically set the regex match to <code>(.*)</code> (to match everything) and the replace string pattern to something like <code>meet.\1</code> which would prefix the meeting code entered into the Virtual Reception with "meet.". You would then configure an associated Call Routing Rule to match calls placed to aliases prefixed with <code>meet.</code> which then strips off that prefix (to leave just the meeting code again) before directing the call to Google Meet.</p>
Post-lookup regex replace string	An optional regular expression used in conjunction with the Post-lookup regex match field to transform the meeting code into a distinct alias pattern that will match a Call Routing Rule configured to route calls into Google Meet conferences. (Only applies if the post-lookup regex match string is also configured and the entered code matches that regex.)
Microsoft Teams options (when a Virtual Reception type of <i>Microsoft Teams</i> is selected)	
Teams Connector	Select the name of the Teams Connector to use to resolve Teams codes.
Lookup location	Specify the location that contains the Conferencing Nodes (typically Proxying Edge Nodes) that will communicate with the Teams Connector to perform the meeting code verification.
Post-lookup regex match	<p>An optional regular expression used to match against the meeting code entered by the caller into the Virtual Reception. This is typically used in conjunction with the Post-lookup regex replace string to transform the meeting code into a distinct alias pattern that will match a Call Routing Rule configured to route calls into Microsoft Teams conferences.</p> <p>For example, you would typically set the regex match to <code>(.*)</code> (to match everything) and the replace string pattern to something like <code>meet.\1</code> which would prefix the meeting code entered into the Virtual Reception with "meet.". You would then configure an associated Call Routing Rule to match calls placed to aliases prefixed with <code>meet.</code> which then strips off that prefix (to leave just the meeting code again) before directing the call to Microsoft Teams.</p>
Post-lookup regex replace string	An optional regular expression used in conjunction with the Post-lookup regex match field to transform the meeting code into a distinct alias pattern that will match a Call Routing Rule configured to route calls into Microsoft Teams conferences. (Only applies if the post-lookup regex match string is also configured and the entered code matches that regex.)
Advanced options	
Destination alias regex match	<p>An optional regular expression used to match against the alias entered by the caller into the Virtual Reception. If the entered alias does not match the expression, the Virtual Reception will not route the call.</p> <p>If this field is left blank, any entered alias is permitted.</p>

Option	Description
Destination alias regex replace string	An optional regular expression used to transform the alias entered by the caller into the Virtual Reception. (Only applies if a regex match string is also configured and the entered alias matches that regex.) Leave this field blank if you do not want to change the alias entered by the caller.
Maximum inbound call bandwidth (kbps)	Enter a value in this field to limit the bandwidth of media being received by Pexip Infinity from each individual participant dialed in to this Virtual Reception. For more information see Managing and restricting call bandwidth .
Maximum outbound call bandwidth (kbps)	Enter a value in this field to limit the bandwidth of media being sent from Pexip Infinity to each individual participant dialed in to this Virtual Reception. For more information see Managing and restricting call bandwidth .
Conference capabilities	Allows you to limit the media content of calls that are connected via the Virtual Reception service. For more information, see Controlling media capability . Default: <i>Main video + presentation</i> .
Maximum call quality	Controls the maximum call quality for participants connecting to this service: <ul style="list-style-type: none"> ◦ <i>Use global setting</i>: use the global maximum call quality setting. ◦ <i>SD</i>: each participant is limited to SD quality. ◦ <i>HD</i>: each participant is limited to HD (720p) quality. ◦ <i>Full HD (1080p)</i>: allows any endpoint capable of Full HD to send and receive its main video at 1080p. Default: <i>Use global setting</i>
Media encryption	Controls the media encryption requirements for participants connecting to this service. <ul style="list-style-type: none"> ◦ <i>Use global setting</i>: use the global media encryption setting. ◦ <i>Best effort</i>: each participant will use media encryption if their device supports it, otherwise the connection will be unencrypted. ◦ <i>Required</i>: all participants (including RTMP participants) must use media encryption. ◦ <i>No encryption</i>: all H.323, SIP and MS-SIP participants must use unencrypted media. (RTMP participants will use encryption if their device supports it, otherwise the connection will be unencrypted.) Default: <i>Use global setting</i>
Service tag	This optional field lets you assign a unique identifier to this service, which you can then use to track use of the service .
Aliases	
Alias: #1	
Alias	Enter the alias that participants will dial to access the Virtual Reception.
Description	An optional description of the alias. This is useful if the Virtual Reception has more than one alias.
Add another alias	Select this option if you want the Virtual Reception to be accessible by more than one alias. For more information, see Using multiple aliases to access the same service .

3. Ensure that every Virtual Meeting Room and Virtual Auditorium that you want to be accessible from this Virtual Reception has an alias in the form of a number. To do this:

- a. Go to either **Services > Virtual Meeting Rooms** or **Services > Virtual Auditoriums**.
- b. Select the name of the Virtual Meeting Room or Virtual Auditorium.
- c. In the Aliases section at the bottom of the page, enter the number to be used to access this Virtual Meeting Room or Virtual Auditorium from the Virtual Reception.

To allow Skype for Business / Lync clients to use Virtual Receptions, you must ensure that the target VMR is configured with at least one alias in the form of a SIP URI that is routable by the SfB/Lync client. (This is in addition to the digits-only VMR alias used by the Virtual Reception.) Pexip Infinity chooses the first alias in the VMR's configuration that is a valid SIP URI; it also

replaces the selected alias's domain with the **Pexip Infinity domain (for Lync / Skype for Business integration)** (from the relevant system location or global setting as appropriate, if configured).

- d. Select **Save**.

Placing calls via the Pexip Infinity Distributed Gateway

The Pexip Infinity Distributed Gateway ("Infinity Gateway") enables endpoints to make calls to other endpoint devices or systems. This includes calls between devices that use different protocols and media formats, such as SIP and H.323 systems, Skype for Business clients (MS-SIP), and Infinity Connect clients (WebRTC). It also enables you to route calls from VTCs and standards-based endpoints into an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.

Traditional hardware gateways are often expensive and are typically centralized in a single location. This means that remote endpoints making gateway calls have to route media over a WAN or over the internet which is costly and uses a lot of bandwidth.

The software-based Infinity Gateway allows for a very cost-efficient deployment of local gateway/transcoding resources in every location. This can result in an improved user experience because of reduced latency as there is no longer a need to hairpin media back to a centralized datacenter. The other benefit is reduced WAN bandwidth usage — again due to not having to hairpin media. Reduced bandwidth usage allows an enterprise to deploy more video systems without having to upgrade the WAN infrastructure.

For example, you can use the Infinity Gateway to enable:

- users of Infinity Connect clients to place a person-to-person call to a SIP endpoint
- Skype for Business / Lync users in your enterprise to make calls to, and receive calls from, virtually any other type of endpoint

Calls to externally-hosted conferences

The Infinity Gateway can also be used to route calls from VTC systems and standards-based endpoints into an externally-hosted conference. It allows:

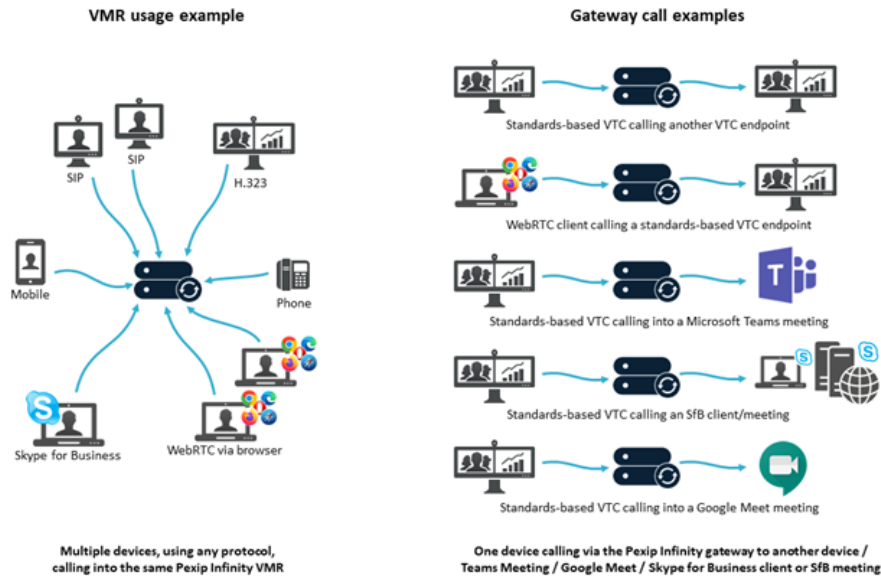
- VTC systems to call directly into a Microsoft Teams or Skype for Business / Lync meeting, or a Google Meet conference
- participants in a Skype for Business / Lync meeting to dial out to (i.e. invite) other non-Skype for Business / Lync participants into the conference.

VMRs versus gateway calls — what's the difference?

Pexip Infinity Virtual Meeting Rooms are used to host multiple participants in the same conference. Those participants could be using a range of different endpoint devices to call into that VMR. In large deployments the VMR conference is likely to be distributed across multiple Conferencing Nodes.

In contrast, a gateway call can be considered as either being placed from one device directly to another device (by dialing the alias associated with that device), or as being placed from one device into an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet (by dialing the address associated with that external conference). As with calls into VMRs, the types of devices being connected (SIP, H.323, WebRTC etc.) does not matter. A single Conferencing Node will typically handle the incoming leg of the call and the outgoing leg, however the call could be distributed across two Conferencing Nodes if required for routing purposes.

Note that the person placing the call may not necessarily know if they are calling into a VMR or making a gateway call, as from their perspective they are typically just dialing an alias. Also, in both cases you could use a Virtual Reception to capture the alias of the VMR or device / external conference that the caller wants to connect into.



How it works

To enable devices to call other devices or systems via the Infinity Gateway, you must configure [Call Routing Rules](#). These rules specify which calls should be interworked, for which protocols, and to where they should be routed.

Incoming calls received by Pexip Infinity are routed as follows:

1. Pexip Infinity receives an incoming call via one of its Conferencing Nodes.
2. It checks whether the destination alias belongs to a Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service; if so, it directs the call to that service.
3. If the alias does not belong to any of the above services, Pexip Infinity checks the Call Routing Rules to see if the alias matches any rules specified there for incoming calls. If so, it places an Infinity Gateway call to the destination alias according to the rule's call target settings (which protocol, location and call control system to use, whether to route to registered devices only, etc).

This means that if an alias matches both a Virtual Meeting Room and a Call Routing Rule, the former will always take precedence and the call will be routed to the Virtual Meeting Room. You must therefore ensure that any regular expressions used in a Call Routing Rule do not unintentionally overlap with any aliases used by a Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service.

The stage where Call Routing Rules are applied in Pexip Infinity's call routing logic for incoming calls is highlighted in the following diagram:



* This information can alternatively be provided by using external or local policy.

† If it is a non-Regular type of Virtual Reception (i.e. Google Meet, Microsoft Teams or SFB/Lync) the Virtual Reception captures the meeting code for the externally-hosted conference, which then needs to be matched by a suitable Call Routing Rule.

- i** If your environment includes a PSTN gateway or uses an ITSP (Internet telephony service provider), consider the potential for toll fraud if you have Call Routing Rules that can route calls to the PSTN gateway or ITSP, or if you allow conference participants to dial out to other participants via the PSTN gateway or ITSP. See [PSTN gateways and toll fraud](#) for more information.

Note that:

- For additional security you can configure rules so that only registered devices are allowed to make calls via the Infinity Gateway.
- By default, the same Conferencing Node that receives the incoming call is used to place the outgoing call. However, you can configure the matching rule to place the call from a Conferencing Node in a specific location. As with all calls, signaling and media may be [handled by different Conferencing Nodes](#) in that location.
- Bandwidth restrictions can be applied to gateway calls; you do this by applying the restriction to the relevant rule.
- If the Infinity Gateway receives DTMF signaling from an inbound call, it will generate similar DTMF on the outbound call.
- In addition to handling gateway calls, rules may also be applied when dialing out from a conference to a new participant (if *Automatic* routing is used). When configuring your rules, consider whether the rule is to apply to incoming gateway calls, outgoing calls from a conference or to both incoming and outgoing calls.

Determining the caller's alias

A gateway call consists of two "legs": the incoming call from the caller to Pexip Infinity, and the outgoing call from Pexip Infinity to the endpoint or meeting being called. In most cases, Pexip Infinity is able to determine the alias of the caller and forward this to the endpoint or meeting being called. However, in some cases callers do not include a domain in their alias (for example, H.323 endpoints can be configured with an alias that does not include a domain), so Pexip Infinity will need to construct a valid alias to use for that caller. It does this by appending one of the following (in order of preference):

- the **Pexip Infinity domain (for Lync / Skype for Business integration)** for the location from which the outbound call is placed,
- the **SIP TLS FQDN** for the Conferencing Node placing the outbound call,
- the **Static NAT address** for the Conferencing Node placing the outbound call,
- the **Secondary interface IPv4 address**, or if this is not configured, the **IPv4 address**, of the Conferencing Node placing the outbound call.

Note that the above does not apply to gateway calls to or from Skype for Business / Lync meetings.

Configuring Call Routing Rules

Call Routing Rules are used to handle two types of calls within Pexip Infinity:

- Calls received by Pexip Infinity that are to be routed via the [Infinity Gateway](#) to their ultimate destination (another device or externally-hosted conference).
- Outgoing calls made from within a Pexip Infinity conference (such as when manually adding a participant to a VMR).

This topic explains [when](#) Call Routing Rules are used, the main [considerations](#) when configuring your rules, our [recommendations](#), how to [add or modify rules](#), and provides an [example](#) rule.

When are Call Routing Rules used?

Call Routing Rules may be used for matching incoming calls received by Pexip Infinity, and they may also be used for matching outbound calls made from a Pexip Infinity conference.

Incoming calls

If an incoming call does not match an alias associated with a conferencing service (Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service), it is deemed a gateway call and it must match a Call Routing Rule for the call to be placed to the destination alias / target system.

When using a Virtual Reception to capture an alias, that new alias is treated as a new incoming call i.e. it will be checked against the service aliases first and then matched against the rules.

- i** Call Routing Rules are **not** required to route incoming calls to a conferencing service (Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service). This happens automatically if the alias received by Pexip Infinity matches an alias associated with the conferencing service.

Outgoing calls from a conference

Outgoing calls made from within a Pexip Infinity conference are checked against Call Routing Rules when:

- Automatically Dialed Participants or participants added manually via the Pexip Infinity Administrator interface have their **Route this call** option set to **Automatically**
- participants are added via an Infinity Connect client.

and then it must match a suitable rule for the call to be placed to the destination alias/target system. See [Recommendations for handling outgoing calls from a conference](#) for more information.

Call routing considerations

When configuring a Call Routing Rule, the main points to consider are:

- **Rule priority:** this is the order in which the rules are checked to see if the conditions specified in the rule match the characteristics of the call that Pexip Infinity is trying to route. Rule checking stops when a match is found, even if the call that is placed as a result of that match fails for any reason.
- **Applicability to incoming or outgoing calls:** you must decide whether the rule applies only to incoming gateway calls, or only to outgoing calls placed from within a Pexip Infinity conference (where **Automatic** routing has been selected), or to both incoming gateway calls and outgoing calls from conferences. You can also limit the rule so that it only applies if the call is being handled by a Conferencing Node in a specific location.
- **Restricting incoming calls to those from registered devices or specific call protocols:** for incoming gateway calls, whether to limit the rule's applicability to only allow calls to be made from devices that are registered to Pexip Infinity, or to limit them to certain incoming protocols e.g. SIP only.
- **Alias matching and transforms:** the dialed alias or alias patterns that apply to this rule. A rule can be configured to apply to a specific alias e.g. `alice@example.com`, or to an alias pattern defined by a regex (regular expression) such as `.*@example.com` (any destination alias with the domain `example.com`). The destination alias can also, optionally, be transformed before the outgoing call is placed.
- **Call media settings:** whether to limit the media capabilities of the call, or whether to apply a theme.
- **Outgoing call placement:** every rule must define to where and how the call is routed to the destination alias. This includes defining the types of devices or systems to which the call is routed, such as limiting it to registered devices only or by specifying the SIP proxy, H.323 gatekeeper or Skype for Business / Lync server to use.
- **Toll fraud:** if you have Call Routing Rules that route calls to an ISDN/PSTN gateway, consider who should be allowed to make those calls (to avoid [toll fraud](#)). You could restrict those rules to only apply to incoming calls from registered devices, or to calls that are being handled in an internal location. Also, to restrict the number of people who can dial out from within a conference, we recommend configuring a Host PIN for your conferences (as only Hosts can initiate an outgoing call).

Note that any incoming call that has a destination alias that matches the alias of any Virtual Meeting Room, Virtual Auditorium, Virtual Reception or Test Call Service is always directed automatically to that conferencing service; the call will not be routed via the Infinity Gateway and the Call Routing Rules are not applied.

This diagram shows the call routing logic within Pexip Infinity when handling incoming call requests and when dialing out from within a conference.



* This information can alternatively be provided by using external or local policy.

† If it is a non-Regular type of Virtual Reception (i.e. Google Meet, Microsoft Teams or SfB/Lync) the Virtual Reception captures the meeting code for the externally-hosted conference, which then needs to be matched by a suitable Call Routing Rule.

Note that as an alternative to using Pexip Infinity's own routing logic, you can configure Pexip Infinity to instead defer its decision making to an external policy service or to use a local policy script (see [policy profiles](#) for more information). This allows Pexip Infinity administrators to implement routing decisions based on their specific requirements.

Recommendations for handling outgoing calls from a conference

- i** If you have any Pexip Smart Scale locations in your deployment, you should not place outgoing calls directly from these locations. See [Placement of outgoing calls from a Pexip Smart Scale location](#) for more information.

In addition to configuring rules to handle person-to-person gateway calls, another consideration when setting up your dial plan is how to deal with outgoing calls placed from within a Pexip Infinity conference, particularly when placed by conference participants using Infinity Connect.

The Infinity Connect clients always use **Automatic** routing when dialing out from a conference, meaning that the user only has to enter or select the alias of the person they want to invite, and they do not have to select a call protocol.

To support the use of **Automatic** routing you **must** configure some appropriate Call Routing Rules otherwise the outbound call will not get placed.

The rules you need to configure depend upon your local requirements, but we recommend that you configure a fallback rule (a rule with a lower priority i.e. a higher number, than all of your other rules that are handling outgoing calls from a conference) that attempts to place the call over SIP and uses either DNS or the local SIP proxy.

Here are some typical rule patterns you may want to use (where all of the rules are configured to handle **Outgoing calls from a conference**):

Priority	Destination alias regex match *	Call target	Protocol and Proxy /Gatekeeper	Purpose
10	.+@example\.com	Registered devices only	n/a	This rule is only required if you are registering endpoints to Pexip Infinity as it will route calls to those registered aliases (devices). You can use a more specific regex if your registered URIs have a more controlled naming convention. You may also want to enable this rule for matching Incoming gateway calls.
190	\d+\.\d+\.\d+\.\d+	Registered device or external system	H.323 (to your gatekeeper or via DNS)	This rule matches an alias in the form of an IP address (see Matching an IP address for a more specific regex) and calls out over H.323 either via your local H.323 gatekeeper or via DNS.
200	(?!.*@example\.com\$).*	Registered device or external system	SIP (to your proxy or via DNS)	<p>This is your fallback rule. It matches aliases that do not end in @example.com (i.e. calls to domains outside of your environment) and routes it out over SIP either via your local SIP proxy or via DNS. To ignore multiple domains, you could use an expression formatted in the style:</p> <pre>(?!.*@(example\.com another\.com)\$).*</pre> <p>Note that these example expressions do not produce any match groups and do not require a replace string (hence the dialed alias is passed on unchanged if the match is successful).</p> <p>You could also tighten the match to only allow certain address formats for the external addresses such as name@domain i.e.</p> <pre>(?!.*@example\.com\$).+@.+</pre>

* All of these rules are intended as examples and use a local video domain of **example.com**. They are a basis for your own requirements and they should be adapted accordingly for your local environment.

Note that Call Routing Rules are **not** required to route incoming calls to a conferencing service (Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service).

Creating and modifying Call Routing Rules

To add, edit or delete a Call Routing Rule, go to **Services > Call Routing**. When specifying a Call Routing Rule, the options are:

Option	Description
Name	<p>The name used to refer to this rule.</p> <p>i If you are using a Virtual Reception as an IVR gateway to capture a conference ID, and then using this Call Routing Rule to transfer the participant into an external conference such as Google Meet, or a Microsoft Teams or Skype for Business / Lync meeting, then the rule Name entered here is shown to conference participants as they are transferred into the meeting (it is overlaid onto the <code>virtual_reception_connecting</code> splash screen of the theme associated with the Virtual Reception that is transferring the call).</p>
Service tag	This optional field lets you assign a unique identifier to this service, which you can then use to track use of the service .
Description	An optional description of the rule.
Priority	<p>Assign a relative priority to this rule, from 1 to 200. Rules are checked in order of priority, starting with 1 and working down the list until a match is found.</p> <p>Rule checking stops when a match is found, even if the call that is placed as a result of that match fails for any reason.</p>
Use this rule for...	
Incoming gateway calls	<p>Applies this rule to incoming calls that have not been routed to a Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service, and thus should be routed via the Infinity Gateway.</p> <p>Default: selected.</p>
Outgoing calls from a conference	<p>Applies this rule to outgoing calls placed from a conferencing service (e.g. when a participant is added to a Virtual Meeting Room) where <i>Automatic</i> routing has been selected.</p> <p>Default: not selected.</p> <p>Note that the same rule can be applied to both incoming and outgoing calls if required.</p>
Calls being handled in location	<p>You can select a specific location, which means:</p> <ul style="list-style-type: none"> for incoming calls, the rule is only applied if the call is being handled (for signaling purposes) by a Conferencing Node in the selected location. for outgoing calls, the rule is only applied if the call is being initiated from the selected location. <p>This allows you to restrict the types of calls that some users can make. For example, you may use a call control system that routes all calls into Pexip Infinity through Conferencing Nodes in an "internal/trusted" location, whereas calls received from devices in the public internet could all be routed to nodes in a different "public/untrusted" location. You can then, for example, use the Calls being handled in location option to allow only calls received in the "trusted" location to place calls via a PSTN gateway.</p> <p>To apply the rule regardless of the location, select <i>Any Location</i>.</p> <p>Default: <i>Any Location</i>.</p>
When matching incoming Gateway calls...	

Option	Description
Match incoming calls from registered devices only	<p>Select this option if you want this rule to only apply to incoming calls from devices, videoconferencing endpoints, soft clients or Infinity Connect clients that are registered to Pexip Infinity. Note that:</p> <ul style="list-style-type: none">The incoming call must be received by the same Conferencing Node as to which the device is registered. This is the normal behavior for most endpoints, however you may need to ensure that the SIP Proxy and Registrar are both configured to resolve to the same node.The call must also match one of the selected protocols below. <p>Calls placed from non-registered clients or devices, or from the Infinity Connect web app will not be routed by this rule if it is enabled.</p> <p>Default: disabled.</p>
Match Infinity Connect (WebRTC/RTMP)	<p>Select the source / protocols of the incoming call to which the rule should apply. Note that selecting Match Lync / Skype for Business (MS-SIP) calls does not apply if Match incoming calls from registered devices only is selected.</p> <p>Default: these options are all enabled by default.</p>
Match SIP	
Match Lync / Skype for Business (MS-SIP)	
Match H.323	
Alias match and transform	
Match against full alias URI	<p>This setting is for advanced use cases and will not normally be required.</p> <p>By default, Pexip Infinity matches against a parsed version of the destination alias, i.e. it ignores the URI scheme, any other parameters, and any host IP addresses. So, if the original alias received by Pexip Infinity is <code>sip:alice@example.com;transport=tls</code> for example, then by default the rule will match against <code>alice@example.com</code>.</p> <p>Select this option to match against the full, unparsed alias instead.</p> <p>You must select this option if you are using this rule to support your Skype for Business / Lync IVR gateway (where this rule is routing calls handled by a Virtual Reception into the relevant Skype for Business / Lync meeting). See Configuring Pexip Infinity as a Skype for Business / Lync gateway for more information.</p>
Destination alias regex match	<p>The regular expression that the destination alias (the alias that was dialed) is checked against to see if this rule applies to this call.</p> <p>Pexip Infinity supports case-insensitive Perl-style regular expression patterns. Note that the regex must match the entire alias — a partial match is treated as a non-match. See Regular expression (regex) reference for information about writing regular expressions.</p>
Destination alias regex replace string	<p>The regular expression string used to transform the originally dialed alias (if a match was found).</p> <p>If you do not want to change the alias, leave this field blank.</p>
Call media settings	
Maximum inbound call bandwidth (kbps)	<p>Enter a value in this field to limit the bandwidth of media being received by Pexip Infinity from each individual participant dialed in via this rule. For more information see Managing and restricting call bandwidth.</p>
Maximum outbound call bandwidth (kbps)	<p>Enter a value in this field to limit the bandwidth of media being sent from Pexip Infinity to each individual participant dialed out from this rule. For more information see Managing and restricting call bandwidth.</p> <p>When dialing out from a conference, the outbound call bandwidth limit is inherited from the VMR's bandwidth settings. In this case, the rule's outbound bandwidth setting configured here is not used.</p>
Call capability	<p>Allows you to limit the media content of the call. The participant being called will not be able to escalate beyond the selected capability. For more information, see Controlling media capability.</p> <p>Default: <i>Main video + presentation</i>.</p>

Option	Description
Maximum call quality	<p>Controls the maximum call quality for participants connecting to this service:</p> <ul style="list-style-type: none"> • <i>Use global setting</i>: use the global maximum call quality setting. • <i>SD</i>: each participant is limited to SD quality. • <i>HD</i>: each participant is limited to HD (720p) quality. • <i>Full HD (1080p)</i>: allows any endpoint capable of Full HD to send and receive its main video at 1080p. <p>Default: <i>Use global setting</i></p>
Media encryption	<p>Controls the media encryption requirements for participants connecting to this service.</p> <ul style="list-style-type: none"> • <i>Use global setting</i>: use the global media encryption setting. • <i>Best effort</i>: each participant will use media encryption if their device supports it, otherwise the connection will be unencrypted. • <i>Required</i>: all participants (including RTMP participants) must use media encryption. • <i>No encryption</i>: all H.323, SIP and MS-SIP participants must use unencrypted media. (RTMP participants will use encryption if their device supports it, otherwise the connection will be unencrypted.) <p>Default: <i>Use global setting</i></p>
Theme	<p>The theme to use for calls placed using this rule. See Themes used by Call Routing Rules (gateway calls) for more information.</p> <p>Default: <i><use Default theme></i> (the global default theme is used).</p>
Outgoing call placement (individual fields are only displayed when appropriate)	
Call target	<p>The device or system to which the call is routed. The options are:</p> <ul style="list-style-type: none"> • <i>Registered device or external system</i>: route the call to a matching registered device if it is currently registered, otherwise attempt to route the call via an external system such as a SIP proxy, Skype for Business / Lync server, H.323 gatekeeper or other gateway/ITSP. • <i>Registered devices only</i>: route the call to a matching registered device only (providing it is currently registered). • <i>Lync / Skype for Business meeting direct (Conference ID in dialed alias)</i>: route the call via a Skype for Business / Lync server to a Skype for Business / Lync meeting. Note that the destination alias must be transformed into just a Skype for Business / Lync Conference ID. • <i>Lync / Skype for Business clients, or meetings via a Virtual Reception</i>: route the call via a Skype for Business / Lync server either to a SfB/Lync client, or — for calls that have come via a Virtual Reception — to a Skype for Business / Lync meeting. For Skype for Business / Lync meetings via Virtual Reception routing, ensure that <i>Match against full alias URI</i> is selected and that the <i>Destination alias regex match</i> ends with <i>.*</i> • <i>Google Meet meeting</i>: routes the call to a Google Meet meeting. See Introduction for more information. • <i>Microsoft Teams meeting</i>: routes the call to a Microsoft Teams meeting. See Integrating Microsoft Teams with Pexip Infinity for more information. • <i>Epic Telehealth Profile</i>: used for telehealth calls. See Epic telehealth integration with Pexip Infinity for more information.
Outgoing location	<p>When calling an external system, this forces the outgoing call to be handled by a Conferencing Node in a specific location.</p> <p>When calling a Skype for Business / Lync meeting, a Conferencing Node in this location will handle the outgoing call, and — for <i>Lync / Skype for Business meeting direct</i> targets — perform the Conference ID lookup on the SfB/Lync server.</p> <p>Select <i>Automatic</i> to allow Pexip Infinity to automatically select the Conferencing Node to use to place the outgoing call (which will usually be the node that received the call).</p> <p>You should never select a Pexip Smart Scale location as the Outgoing location.</p>

Option	Description
Protocol	<p>The protocol used to place the outgoing call. Note that:</p> <ul style="list-style-type: none"> If the call is to a registered device, Pexip Infinity will instead use the protocol that the device used to make the registration. <i>RTMP (streaming)</i> always routes to an external system. <p>Default: <i>SIP</i>.</p>
SIP Proxy *	<p>You can optionally specify the SIP Proxy to use to place an outgoing SIP call.</p> <p>Default: <i>Use DNS</i>.</p>
H.323 Gatekeeper *	<p>You can optionally specify the H.323 Gatekeeper to use to place an outgoing H.323 call.</p> <p>Default: <i>Use DNS</i>.</p>
Lync / Skype for Business server *	<p>When calling an external system, this is the Skype for Business / Lync server to use for outbound SfB/Lync (MS-SIP) calls.</p> <p>When calling a Skype for Business / Lync meeting, this is the SfB/Lync server to use for the Conference ID lookup and to place the call.</p> <p>Default: <i>Use DNS</i> (note that a server must be selected when the Call target is <i>Lync / Skype for Business meeting direct (Conference ID in dialed alias)</i>).</p>
TURN server †	<p>You can optionally specify the TURN server to use when placing calls to ICE-enabled clients (such as Skype for Business / Lync clients and Infinity Connect WebRTC clients).</p>
STUN server †	<p>You can optionally specify the STUN server to use when placing calls to ICE-enabled clients (such as Skype for Business / Lync clients and Infinity Connect WebRTC clients).</p>
Teams Connector	<p>You can optionally select the Teams Connector you want to handle the call. If you do not specify anything, the Teams Connector associated with the outgoing location is used.</p>
Access token	<p>Select the name of the access token to use to resolve Google Meet IDs. You should select either a trusted or untrusted type of token, depending on whether you want to enable the device to be automatically admitted into the Google Meet conference (subject to also being a trusted endpoint from Pexip Infinity's perspective i.e. if the rule also has Treat as trusted enabled).</p> <p>Typically, you will use a trusted token if Treat as trusted is selected, and an untrusted token if Treat as trusted is not selected.</p>
Treat as trusted	<p>This indicates that the target of this Call Routing Rule may treat the caller as part of the target organization for trust purposes. It can be applied to calls placed to:</p> <ul style="list-style-type: none"> Microsoft Teams Google Meet SIP destinations and registered SIP devices (via a P-Asserted-Identity in the SIP header)
External participant avatar lookup	<p>Applies to Microsoft Teams integrations only. This determines whether or not the Teams Connector requests from Exchange Online an avatar for each participant in the Teams conference.</p> <ul style="list-style-type: none"> <i>Use global setting</i>: use the global external participant avatar lookup setting. <i>Yes</i>: request the participant avatar from Exchange Online via the Teams Connector. <i>No</i>: use default avatar behavior. <p>Default: <i>Use global setting</i></p>
Rule state	

Option	Description
Enable this rule	Determines if the rule is enabled or not. Any disabled rules still appear in the rules list but are ignored. Use this setting to test configuration changes, or to temporarily disable specific rules.

* If you do not select a specific **H.323 Gatekeeper**, **SIP Proxy** or **SfB/Lync server**, the Conferencing Node will attempt to use DNS to locate an appropriate system via which to route the call (rather than fall back to using the gatekeeper / proxy / server associated with the node's system location).

† If you do not select a specific **TURN server** or **STUN server**, the call will use the TURN / STUN server associated with the node's system location.

Example

In our example organization, every employee has their own video endpoint that is registered to Pexip Infinity with an alias in the format `firstname.lastname@example.com`, and their own Virtual Meeting Room with an alias in the format `meet.firstname.lastname@example.com`.

In most cases, employees will use their standalone SIP or H.323 endpoints to call others within the organization, but sometimes they want to use Infinity Connect to make a person-to-person call.

To support this, we set up the following Call Routing Rule (unspecified settings can be left as their default values):

Option	Input	Notes
Name	Infinity Connect to SIP	
Description	Allow Infinity Connect users to call registered SIP endpoints directly	
<input checked="" type="checkbox"/> Incoming gateway calls <input type="checkbox"/> Outgoing calls from a conference		We want this rule to match incoming gateway calls only.
Calls being handled in location	<i>Any location</i>	We want this rule to apply throughout our deployment.
<input checked="" type="checkbox"/> Match Infinity Connect (WebRTC / RTMP) <input type="checkbox"/> Match SIP <input type="checkbox"/> Match Lync / Skype for Business (MS-SIP) <input type="checkbox"/> Match H.323		In this example, we only want the rule to apply to calls from Infinity Connect clients.
Destination alias regex match	<code>.*@example.com</code>	This regular expression will match any destination alias with the domain <code>example.com</code> .
Destination alias regex replace string	<blank>	We have left this blank because we do not want to amend the alias.
Call target	<i>Registered devices only</i>	We want the call to match currently registered devices only.

This rule means that if an Infinity Connect user dials any alias with the domain `@example.com` (e.g. `alice.jones@example.com`), the call will be routed over SIP. We do not need to worry about this rule applying to VMR aliases with the same domain (e.g. `meet.alice.jones@example.com`) because rules are only applied to incoming calls after checking whether there are any VMRs, Virtual Receptions etc. with that alias.

Configuring the Test Call Service

Pexip Infinity provides a test loopback service that allows users to check the quality of their video and audio (i.e. that their local camera, microphone and speakers are working properly), and verifies that they can connect to a Conferencing Node.

Test Call Services are configured within Pexip Infinity in the same manner as other services such as Virtual Reception. By default, Pexip Infinity includes a preconfigured Test Call Service (called "Test Call Service") and an associated alias of "test_call" that can be used to dial into the Test Call Service. Additional test call services and aliases can be added if required.

We recommend that you add a `test_call@<yourdomain>` alias to the preconfigured Test Call Service to cater for devices that automatically add a domain to their called alias.

After a call into a Test Call Service (e.g. to the "test_call" alias) is answered, the user receives some instructions and then the user's video and audio of themselves is played back from the Conferencing Node with a 2 second delay. The test call then automatically disconnects after approximately 20 seconds.

The instructions and timeouts can all be [customized](#) via themes, and different themes can be applied to different Test Call Services, if required.

As an administrator, you also have the ability to [dial out](#) to a user (alias) to initiate a test call from within Pexip Infinity.

Configuring Test Call Services and aliases

To configure Test Call Services and their associated aliases, go to **Services > Test Call Service**. The options are:


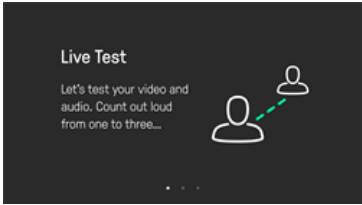
Option	Description
Name	The name used to refer to this Test Call Service.
Description	A description of the Test Call Service.
Theme	The theme for use with this Test Call Service. For more information, see Customizing conference images and voice prompts using themes . Default: <code><use Default theme></code> (the global default theme is used).
Advanced options	
Maximum inbound call bandwidth (kbps)	Enter a value in this field to limit the bandwidth of media being received by Pexip Infinity from the user dialed in to this Test Call Service. For more information see Managing and restricting call bandwidth .
Conference capabilities	Allows you to limit the media content of the conference. For more information, see Controlling media capability . Default: <i>Main video + presentation</i> .
Maximum call quality	Controls the maximum call quality for participants connecting to this service: <ul style="list-style-type: none"> <i>Use global setting</i>: use the global maximum call quality setting. <i>SD</i>: each participant is limited to SD quality. <i>HD</i>: each participant is limited to HD (720p) quality. <i>Full HD (1080p)</i>: allows any endpoint capable of Full HD to send and receive its main video at 1080p. Default: <i>Use global setting</i>
Media encryption	Controls the media encryption requirements for participants connecting to this service. <ul style="list-style-type: none"> <i>Use global setting</i>: use the global media encryption setting. <i>Best effort</i>: each participant will use media encryption if their device supports it, otherwise the connection will be unencrypted. <i>Required</i>: all participants (including RTMP participants) must use media encryption. <i>No encryption</i>: all H.323, SIP and MS-SIP participants must use unencrypted media. (RTMP participants will use encryption if their device supports it, otherwise the connection will be unencrypted.) Default: <i>Use global setting</i>
Service tag	This optional field lets you assign a unique identifier to this service, which you can then use to track use of the service .
Aliases	
Alias: #1	

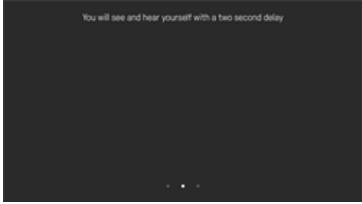
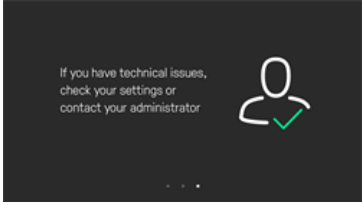
Option	Description
Alias	<p>The alias that, when received by Pexip Infinity, will cause it to route the call to this service.</p> <p>The alias entered here must match the alias as it is received by Pexip Infinity. Wildcards and regular expressions are not supported.</p> <p>In most cases, the alias received by Pexip Infinity will be the same as the alias that the conference participant dialed from their endpoint, but there are some exceptions, described in About aliases and access numbers.</p> <p>You may also want to define multiple aliases for the same service to ensure that it can be accessed by devices and protocols that enforce specific alias formats — for more information, see Using multiple aliases to access the same service.</p>
Description	An optional description of the alias. This is useful if you have more than one alias for a service. Note that this description may be displayed to end users on registered Infinity Connect clients who are performing a directory search.
Add another Alias	Select this option if you want the Test Call Service to be accessible by more than one alias. For more information, see Using multiple aliases to access the same service .

Customizing the Test Call Service via themes

As with other Pexip Infinity services, the images that are shown, and the audio prompts that the end-user hears, can all be fully customized via [themes](#). You can also control the playback delay and the duration of the test call by configuring settings in the `themeconfig.json` theme file.

The following theme files, splash screens and settings are used during a test call:

Theme file/setting	Default content	Notes
conf-test_call_48kHz_mono.wav	"Let's test your video and audio. Count out loud from one to three, now."	Audio file played at the start of a video call to a Test Call Service.
conf-test_call_audio_only_48kHz_mono.wav	"Let's test your audio settings. Count out loud from one to three, now."	Audio file played at the start of an audio-only call to a Test Call Service.
conf-test_call_disconnect_48kHz_mono.wav	"If you have technical issues, check your settings or contact your administrator."	Audio file played at the end of a call to a Test Call Service.
background_test_call.jpg		The background image (a black screen) used by default on the Test Call Service splash screens.
test_call_welcome splash screen		<p>Shown at the start of a call to a Test Call Service.</p> <p>Theme elements used:</p> <ul style="list-style-type: none"> Icon: <code>icon_test_call_welcome.svg</code> Label: <code>test_call_welcome_header</code> and <code>test_call_welcome_text</code> Background: <code>background_test_call.jpg</code>

Theme file/setting	Default content	Notes
test_call_in_progress splash screen		Shown during a call to a Test Call Service. Note that a large, live (with a short delay) video image of the test call participant is shown on top of this screen during a test call. Theme elements used: <ul style="list-style-type: none"> Label: test_call_in_progress Background: background_test_call.jpg
test_call_complete splash screen		Shown briefly prior to automatically disconnecting the participant from a Test Call Service. Theme elements used: <ul style="list-style-type: none"> Icon: icon_test_call_complete.svg Label: test_call_complete Background: background_test_call.jpg
test_call_service_media_delay (in themeconfig.json)	2	The number of seconds that media is delayed before being looped back to the caller when using a Test Call Service.
test_call_service_disconnect_timeout (in themeconfig.json)	10	The number of seconds that a user can test their media before the disconnect message is played, when using a Test Call Service.

The following sequence describes the test call process and explains when each theme file and setting is used:

1. When a test call is answered, the `test_call_welcome` splash screen is displayed and either the `conf-test_call_48kHz_mono.wav` or `conf-test_call_audio_only_48kHz_mono.wav` audio file is played, depending on whether the caller is connecting with video or just with audio-only.
2. The `test_call_in_progress` splash screen is displayed (after the `conf-test_call_48kHz_mono.wav` audio file has finished).
3. The caller's audio and video media is replayed back to them with a `<test_call_service_media_delay>` seconds delay (2 seconds by default).
4. The media replay stops after `<test_call_service_disconnect_timeout>` seconds (10 seconds by default).
5. The `test_call_complete` splash screen is displayed and the `conf-test_call_disconnect_48kHz_mono.wav` audio file is played.
6. The call automatically disconnects after a further 5 seconds (not configurable).

Dialing out from the Test Call Service

Typically, users will **dial in** to the Test Call Service to check their video and audio, but as an administrator you can **dial out** to a user (alias) to initiate a test call from within Pexip Infinity. When the user answers the call they are taken through the test in the same way as if they had dialed the Test Call Service themselves.

To dial a user into a test call:

1. Select the Test Call Service to dial the participant from (go to **Services > Test Call Service** and select a service).
2. At the bottom left of the screen, select **Dial out to participant**.
3. Enter the **Participant alias** you want to dial, and the **Protocol** to use to make the call.
4. Select **Dial out to participant**.

Registering devices to Pexip Infinity

Pexip Infinity can act as a SIP registrar and H.323 gatekeeper, which means that you can register SIP and H.323 endpoints directly to Pexip Infinity. This allows Pexip Infinity to route calls to those registered devices without having to go via an external SIP proxy or H.323 gatekeeper, or rely on DNS.

Infinity Connect desktop clients and legacy versions of the Infinity Connect mobile clients for Android can also register to Pexip Infinity Conferencing Nodes. This allows these devices to receive calls via Pexip Infinity and use directory lookup services.

Devices can only register to Pexip Infinity with a permitted alias and by supplying valid credentials (if authentication is required). Allowed aliases and their associated credentials can be configured manually, or they can be [bulk provisioned](#) from directory information contained in a Windows Active Directory LDAP server, or any other LDAP-accessible database.

Additionally, Infinity Connect desktop clients support [Active Directory \(AD\)](#) Single Sign-On (SSO) services, meaning users can register to Pexip Infinity and authenticate their clients using their AD credentials.

Note that devices do not need to be registered in order to make calls to Pexip Infinity. However, you can configure your deployment so that only registered devices can make [gateway calls](#), thus preventing potential for toll fraud.

For instructions on how to register the Infinity Connect client, see [Registering and provisioning the Infinity Connect client](#).

Configuration summary for enabling registrations

To configure Pexip Infinity to accept registrations and route calls to registered devices you must:

1. Ensure that the Pexip Infinity's registrar service is enabled (**Services > Registrar**).
2. Configure the aliases that devices are allowed to register to Pexip Infinity (**Users & Devices > Device Aliases**).
3. Consider if authentication is required, what form that authentication should take, and whether to provision individual users with their registration details, and then configure the appropriate services and credentials.
4. Configure appropriate Call Routing Rules (**Services > Call Routing**). Rules are required in all cases except for when using **Manual** routing to add a participant to a conference.

Each step is described in more detail below, after the explanation of how Pexip Infinity manages registered devices.

How it works

Registering devices

The registrar service on Pexip Infinity is enabled by default. However, a device can only register to Pexip Infinity if the alias it wants to register has been added to Pexip Infinity's list of allowed device aliases (**Users & Devices > Device Aliases**).

Device registrations can optionally also be subject to authentication. To enforce authentication, username and password credentials must be specified for each alias that is added to Pexip Infinity's list of allowed device aliases (unless you are using Infinity Connect clients with SSO services). When credentials are specified, the device's registration request is challenged and the device is only allowed to register with that alias if it provides matching credentials. Pexip Infinity supports Digest authentication only.

A device can register to any Conferencing Node. Pexip Infinity supports H.323 alternate gatekeeper registrations for nodes in the same system location, but does not apply any other form of registration load balancing or failover. Multiple devices can register with the same alias. After a device has registered it must periodically refresh that registration.

SIP and H.323 devices

You can register SIP and H.323 devices to Pexip Infinity:

- To register a **SIP device**, use the IP address or FQDN of a local Conferencing Node as the SIP proxy.
- To register an **H.323 device**, use the IP address or FQDN of a local Conferencing Node as the H.323 gatekeeper.
All of the device aliases presented in an H.323 registration request must be in the list of allowed device aliases. If any alias is not present, none of the aliases will be allowed to register. Additionally, if device authentication is being used, all of the device aliases in the request must be configured with the same credentials.

If your endpoint supports both SIP and H.323, you should register it to Pexip Infinity over just one protocol, either SIP or H.323, but not both.

Infinity Connect registrations

To register an **Infinity Connect client**, ensure that either the IP address or FQDN of a local Conferencing Node is used as the configured **Registration Host** address. Alternatively, you can specify a domain if you have [set up the appropriate DNS records](#). Infinity Connect clients register over the WebRTC protocol.

- Infinity Connect clients can register in one of two ways: **non-SSO**, which can optionally include a username and password for authentication, or **using SSO**, which uses a Single Sign On service (SSO) such as [AD FS](#) for authentication. For any given alias, we recommend that you enable Infinity Connect registrations for just one of these methods, not both. This is particularly important if you have enabled non-SSO registrations but have not also configured username and password authentication credentials.
- You can provision individual users with their registration details and automatically apply those registration settings to their Infinity Connect client. See [Registering and provisioning the Infinity Connect client](#) for more information.
- When your Infinity Connect client is registered, as well as being able to receive calls, you can filter and lookup the contact details (phone book / directory) of other devices or VMRs that are set up on the Pexip Infinity platform. See [Directory \(phone book\) of devices and VMRs for registered Infinity Connect clients](#) for more information.

Calling registered devices

Whenever Pexip Infinity needs to place a call it follows the [Call routing logic](#) shown in the diagram below. If the alias it is trying to place the call to is currently registered, then Pexip Infinity will place the call to that registered device instead of attempting to find it via other means such as call control or DNS.

When calling a registered device:

- the Conferencing Node to which the device is registered will place the call to the device (media may flow through other nodes, as per the platform's standard distributed conferencing behavior)
- the outbound call is always placed to the registered device over the same protocol that it used to make the registration
- if multiple devices are registered with the same alias, Pexip Infinity will place a call to each device, i.e. it will fork the call.

Requirement for Call Routing Rules

In most cases, you must also configure a suitable Call Routing Rule that instructs Pexip Infinity to place the outbound call to the device. A Call Routing Rule **is** required when:

- making a call to another device or system via the Infinity Gateway
- using **Automatic** routing when adding a participant to a conference.

A Call Routing Rule **is not** required when:

- using **Manual** routing when adding a participant to a conference.

When configuring rules you must define:

- whether the rule applies only to incoming gateway calls, or only to outgoing calls placed from within a Pexip Infinity conference (where **Automatic** routing has been selected), or to both incoming gateway calls and outgoing calls from conferences
- the destination aliases to be matched — this needs to match the pattern of the registered device aliases
- whether to route the call to **Registered devices only** (in which case the call will fail if the device is not currently registered), or to **Registered device or external system** which will route the call to a matching registered device if it is currently registered, otherwise it will attempt to route the call via an external system.

Avoiding call looping

Call Routing Rules that are intended to route calls to devices registered to Pexip Infinity (e.g. Infinity Connect clients) could, depending upon DNS configuration, result in call looping if the destination device alias is not currently registered.

This can occur if the Call Routing Rule handling those calls has its **Call target** set to **Registered device or external system**. In this case, if the alias is not currently registered, Pexip Infinity will attempt to place the call via DNS or a call control system. This could result in the call being routed back to Pexip Infinity which would then match the same Call Routing Rule and result in a loop.

To avoid this, we recommend that you configure a **Call target** of **Registered devices only** for your Call Routing Rules that match those device aliases that you expect to be registered. This ensures that the call will fail cleanly without looping if the device is not currently registered.

Note that in earlier versions of Pexip Infinity before the availability of the **Registered devices only** option, we previously recommended configuring a SIP proxy with an address of **nowhere.invalid**. This workaround will still avoid the call looping issue, but if you have any existing rules that use this method, we recommend modifying them to use a **Call target** of **Registered devices only** instead.

Call routing logic

The stage in Pexip Infinity's call routing logic where it checks whether the device that is being called is registered is highlighted in the following diagram:



* This information can alternatively be provided by using external or local policy.

† If it is a non-Regular type of Virtual Reception (i.e. Google Meet, Microsoft Teams or SfB/Lync) the Virtual Reception captures the meeting code for the externally-hosted conference, which then needs to be matched by a suitable Call Routing Rule.

Note that you can configure rules so that only registered devices are allowed to make calls via the Infinity Gateway.

Configuring the registrar service

Devices can only register to Pexip Infinity if the registrar service is enabled.

To configure Pexip Infinity's registrar service, go to **Services > Registrar**. The options are:

Option	Description
Enable registrar	Controls whether devices can register to Pexip Infinity. Devices must register with a permitted alias (configured via Users & Devices > Device Aliases). The registrar service is enabled by default.
Registration refresh (general)	
Registration refresh strategy	Defines which strategy to use when calculating the expiry time of a SIP or H.323 registration: <ul style="list-style-type: none"> Adaptive: Pexip Infinity automatically adjusts the refresh interval depending on the number of current registrations on the Conferencing Node handling the registration request, in order to spread the load of registration refreshes. As the number of devices registered to a Conferencing Node increases, the refresh interval calculated by Pexip Infinity will also typically increase, although it will always be within the bounds of the configured minimum and maximum refresh intervals. Basic: Pexip Infinity simply uses the configured minimum and maximum settings, along with the requested value, to determine the refresh interval. Default: Adaptive.
Minimum refresh interval (Adaptive strategy)	The minimum interval in seconds before a device's registration must be refreshed, when using the Adaptive strategy. Default: 60 seconds.
Maximum refresh interval (Adaptive strategy)	The maximum interval in seconds before a device's registration must be refreshed, when using the Adaptive strategy. Default: 3600 seconds.
Minimum refresh interval (Basic strategy)	The minimum interval in seconds before a device's registration must be refreshed, when using the Basic strategy. Default: 60 seconds.
Maximum refresh interval (Basic strategy)	The maximum interval in seconds before a device's registration must be refreshed, when using the Basic strategy. Default: 300 seconds.
Registration refresh intervals for SIP endpoints behind NATs	
Minimum refresh interval (when behind NAT)	The minimum interval in seconds before a device's registration must be refreshed, for a SIP endpoint behind a NAT. Default: 60 seconds.
Maximum refresh interval (when behind NAT)	The maximum interval in seconds before a device's registration must be refreshed, for a SIP endpoint behind a NAT. The refresh interval for SIP endpoints behind a NAT typically has to be shorter than the interval for other endpoints. This is to help keep the NAT pinhole alive. Default: 90 seconds.
Call routing for desktop clients	
Route calls via registrar	When enabled, all calls from registered Infinity Connect desktop clients are routed via the Conferencing Node to which the client is registered, regardless of the domain being called. The client uses the previously-discovered IP address of the Conferencing Node — it does not perform a DNS SRV lookup of the Registration Host server address each time a call is placed. When disabled, DNS is used to identify the Conferencing Node to which calls are placed from registered clients. Default: enabled.

Note that:


- In all circumstances, requests for a value lower than the relevant **Minimum refresh delay** will result in the registration being rejected.
- The registration refresh strategies apply only to SIP and H.323 endpoints. Infinity Connect clients have a fixed refresh interval of 120 seconds.
- Any changes to the **Route calls via registrar** setting are picked up by the client each time it re-registers or refreshes its registration.

Specifying the aliases that devices are allowed to register with

Devices can only register to Pexip Infinity with a permitted alias.

You can [bulk provision](#) the device aliases from directory information contained in a Windows Active Directory LDAP server, or any other LDAP-accessible database. You can also import device aliases using a [CSV file](#).

To manually configure the aliases that devices are allowed to register to Pexip Infinity, go to **Users & Devices > Device Aliases**. The options are:

Option	Description
Device alias	The alias URI that a device/client can register to Pexip Infinity. The alias must be an exact match; regular expressions are not supported. It cannot be blank.
Creation time	This read-only field shows the date and time when this record was first configured.
Service tag	This optional field lets you assign a unique identifier to this service, which you can then use to track use of the service .
Description	An optional description of the device. Note that this description may be displayed to end users on registered Infinity Connect clients who are performing a directory search.
Enable SIP registration	Allows this device alias to register over the SIP protocol. The registration is optionally authenticated using the specified Username and Password .
Enable H.323 registration	Allows this device alias to register over the H.323 protocol. The registration is optionally authenticated using the specified Username and Password .
Enable Infinity Connect registration (non-SSO)	Allows an Infinity Connect client to register using this alias (not using SSO services). The registration is optionally authenticated using the specified Username and Password .
Enable Infinity Connect registration using SSO	Allows an Infinity Connect client to register using this alias, using Single Sign-On (SSO) services such as AD FS to authenticate the registration. For more information, see Infinity Connect registrations .
Username and Password	These credentials apply to aliases that are enabled to register using SIP , H.323 or Infinity Connect non-SSO registration . They are used to authenticate the device's registration. Credentials are optional and the device is not challenged if no credentials are entered.  The username and password are case-sensitive.
Device origin	The name of the LDAP sync template used to create this device alias (it is blank if the device was created by manual input or via the API). This field is read-only.
Owner's email address	The email address of the owner of the device alias. Provisioning messages for this device alias will be sent to this address. This field is required for devices registering using Infinity Connect registration using SSO .

Directory (phone book) of devices and VMRs for registered Infinity Connect clients

Pexip Infinity provides a directory search facility to all Infinity Connect clients that are registered to a Conferencing Node.

Upon request from the Infinity Connect client (typically when the user performs a search on their contact list), Pexip Infinity will return a list of device and conference aliases that match the search string entered by the user.

Pexip Infinity checks the supplied search string to see if it is contained within the alias or alias description of any of its configured services (Virtual Meeting Rooms, Virtual Auditoriums or Virtual Receptions), or within the device aliases/descriptions that are allowed to register to Pexip Infinity (it does not matter if that device alias is currently registered or not, and no presence information is supplied).

Pexip Infinity returns up to 5 service aliases and up to 5 device aliases.

The directory service can be controlled by the **Enable directory** global setting (**Platform > Global Settings > Connectivity**). It is enabled by default.

Feature extensions via the external policy API


If you make use of the external policy API, the external system can be used to:

- determine whether a device alias is allowed to register to a Conferencing Node.
- obtain extended directory information — this can replace or be used in addition to the list of local device and VMR aliases supplied to registered Infinity Connect clients.

See [Using external and local policy to control Pexip Infinity behavior](#) for more information.

Troubleshooting and logging

To see a list of all device aliases that are currently registered to the Pexip Infinity platform, go to **Status > Registrations**.

-  Each device alias configured in Pexip Infinity has to be enabled for the individual protocols over which that alias can be registered. If the alias is configured but the device cannot register, you should check which protocols have been enabled for registration for that alias.

Successful registration requests and expired registrations are recorded in the administrator log, for example:

```
Message="Registration added" Alias="bob@example.com" Protocol="SIP" Registration-id="958e52ba-4328-424b-8d77-c18a59f4c1da"
Natted="False" Location="Europe"
Message="Registration deleted" Alias="55170" Protocol="H323" Registration-id="302fd156-85bd-497d-85dc-c5b29a16d612"
Location="Europe" Reason="Registration expired"
```

Failed registration requests (due to, for example, an unknown device alias or authorization failure) and refreshed registrations are recorded in the support log only.

If a call fails to be placed to a registered device, check that an appropriate Call Routing Rule has been configured that has suitable destination alias matching strings for the registered alias (and ensure that [call looping](#) cannot occur).

Note that registrations are enabled by default when Pexip Infinity is installed. Therefore, until device aliases are configured, every registration request will be rejected.

Maintenance mode

Devices cannot register to a Conferencing Node that is in maintenance mode:

- Requests from **SIP devices** and **Infinity Connect clients** are rejected with "503 Service Unavailable".
- Requests from **H.323 endpoints** receive no response.
 - If the H.323 endpoint has previously registered and received a list of alternate gatekeepers, it should then attempt to register to one of the alternates instead.
 - If the H.323 endpoint has not previously registered (and therefore has not received a list of alternate gatekeepers) it will typically be unable to register. However, it may attempt to register to another Conferencing Node if multiple h323rs DNS SRV records have been configured and the endpoint can handle multiple records.

SIP registration behavior

The alias registered by a SIP device alias is normally a URI, for example `alice@example.com`, and thus the full device alias (including any domain) must match an entry in the **Device aliases** list.

When troubleshooting failed registrations, you need to look in the **support log**. Typical messages related to various successful and unsuccessful SIP registration attempts are described in the following table:

Registration scenario	Behavior and example support log messages
No authentication is required and the alias exists in the list of device aliases	<p>Register request is accepted with 200 response</p> <pre> Message="Summarised received SIP request" Src-address="10.44.2.77" Src-port="42576" Dst-address="10.44.155.21" Dst-port="5061" Transport="TLS" Method="REGISTER" From="sip:bob@example.com" To="sip:bob@example.com" Contacts="<sip:bob@10.44.2.77>;+sip.instance=<urn:uuid:f0b7095d-5ee0-548a-80a8-c522aafdf94b>" Call-ID="58078da9a56dbcee@10.44.2.77" CSeq="62195" Request-URI="sip:example.com" Received-time="2017-03-23T16:11:59,795756" Message="Registration added" Alias="bob@example.com" Protocol="SIP" Registration-id="cacef4f2-dc7b-4cbe-9a6f-6c69aea640fb" Natted="False" Location="Europe" Message="Summarised sending SIP response" Src-address="10.44.155.21" Src-port="5061" Dst-address="10.44.2.77" Dst-port="42576" Transport="TLS" Method="REGISTER" From="sip:bob@pexip.com" To="sip:bob@example.com" Contacts="<sip:bob@10.44.2.77>;+sip.instance=<urn:uuid:f0b7095d-5ee0-548a-80a8-c522aafdf94b>;expires=74" Call-ID="58078da9a56dbcee@10.44.2.77" CSeq="62195" Status-code="200" </pre>
Requested alias does not exist in the list of device aliases	<p>Register request is rejected with 403 response</p> <pre> Message="Summarised received SIP request" Src-address="10.44.2.77" Src-port="42576" Dst-address="10.44.155.21" Dst-port="5061" Transport="TLS" Method="REGISTER" From="sip:bob@example.com" To="sip:bob@example.com" Contacts="<sip:bob@10.44.2.77>;+sip.instance=<urn:uuid:f0b7095d-5ee0-548a-80a8-c522aafdf94b>" Call-ID="58078da9a56dbcee@10.44.2.77" CSeq="62191" Request-URI="sip:example.com" Received-time="2017-03-23T16:02:29,795311" Message="Summarised sending SIP response" Src-address="10.44.155.21" Src-port="5061" Dst-address="10.44.2.77" Dst-port="42576" Transport="TLS" Method="REGISTER" From="sip:bob@example.com" To="sip:bob@example.com" Contacts="<sip:bob@10.44.2.77>;+sip.instance=<urn:uuid:f0b7095d-5ee0-548a-80a8-c522aafdf94b>" Call-ID="58078da9a56dbcee@10.44.2.77" CSeq="62191" Status-code="403" </pre>
Authentication is required but the wrong credentials are supplied	<p>Register request is rejected with 401 response (two requests and two 401 responses)</p> <pre> Message="Summarised received SIP request" Src-address="10.44.2.77" Src-port="42576" Dst-address="10.44.155.21" Dst-port="5061" Transport="TLS" Method="REGISTER" From="sip:bob@example.com" To="sip:bob@example.com" Contacts="<sip:bob@10.44.2.77>;+sip.instance=<urn:uuid:f0b7095d-5ee0-548a-80a8-c522aafdf94b>" Call-ID="58078da9a56dbcee@10.44.2.77" CSeq="62171" Request-URI="sip:example.com" Received-time="2017-03-23T15:44:17,530041" Message="Summarised sending SIP response" Src-address="10.44.155.21" Src-port="5061" Dst-address="10.44.2.77" Dst-port="42576" Transport="TLS" Method="REGISTER" From="sip:bob@example.com" To="sip:bob@example.com" Contacts="<sip:bob@10.44.2.77>;+sip.instance=<urn:uuid:f0b7095d-5ee0-548a-80a8-c522aafdf94b>" Call-ID="58078da9a56dbcee@10.44.2.77" CSeq="62171" Status-code="401" </pre> <p>(this pair of messages is then repeated)</p>
Authentication is required and correct credentials are supplied	<p>The first register request is rejected with 401 response, then the second request (when the device will supply the requested credentials) is accepted with 200</p> <pre> Message="Summarised received SIP request" Src-address="10.44.2.77" Src-port="42576" Dst-address="10.44.155.21" Dst-port="5061" Transport="TLS" Method="REGISTER" From="sip:bob@example.com" To="sip:bob@example.com" Contacts="<sip:bob@10.44.2.77>;+sip.instance=<urn:uuid:f0b7095d-5ee0-548a-80a8-c522aafdf94b>" Call-ID="58078da9a56dbcee@10.44.2.77" CSeq="62177" Request-URI="sip:example.com" Received-time="2017-03-23T15:48:39,817656" Message="Summarised sending SIP response" Src-address="10.44.155.21" Src-port="5061" Dst-address="10.44.2.77" Dst-port="42576" Transport="TLS" Method="REGISTER" From="sip:bob@example.com" To="sip:bob@example.com" Contacts="<sip:bob@10.44.2.77>;+sip.instance=<urn:uuid:f0b7095d-5ee0-548a-80a8-c522aafdf94b>;expires=111" Call-ID="58078da9a56dbcee@10.44.2.77" CSeq="62178" Request-URI="sip:example.com" Received-time="2017-03-23T15:48:39,873782" Message="Registration added" Alias="bob@example.com" Protocol="SIP" Registration-id="aecae4f2-da7b-4dbe-9a6f-6c69feb689ea" Natted="False" Location="Europe" Message="Summarised sending SIP response" Src-address="10.44.155.21" Src-port="5061" Dst-address="10.44.2.77" Dst-port="42576" Transport="TLS" Method="REGISTER" From="sip:bob@example.com" To="sip:bob@example.com" Contacts="<sip:bob@10.44.2.77>;+sip.instance=<urn:uuid:f0b7095d-5ee0-548a-80a8-c522aafdf94b>;expires=111" Call-ID="58078da9a56dbcee@10.44.2.77" CSeq="62178" Status-code="200" </pre>

H.323 registration behavior

An H.323 device often registers multiple aliases such as an H.323 ID, an E.164 number, and in some cases the system name. H.323 ID aliases can take the form of a full URI.

All of the device aliases presented in an H.323 registration request must be in the list of allowed device aliases. If any alias is not present, none of the aliases will be allowed to register. Additionally, if device authentication is being used, all of the device aliases in the request must be configured with the same credentials.

When an alias does not exist in the list of device aliases, a message is written to the support log in the form:

```
Name="support.registration" Message="H.323 device authentication failed" Reason="Alias not found" Alias="(u'Bob Jones', 'h323_ID')"
```

To check all the aliases that are being presented from an endpoint, you can filter the support log by "registrationRequest", and then search for one of the aliases that you know is being presented. When you have found a suitable message, the `terminalAlias` section lists all of the aliases that are being presented, for example:

```
Name="support.h323.ras" Message="Received RAS message" Src-address="10.44.2.77" Src-port="1719" Dst-address="10.44.157.21" Dst-port="1719" Detail="registrationRequest:
...
terminalAlias: [
h323_ID:
  Bob Jones,
h323_ID:
  bob@example.com
dialedDigits:
  123456
]
```

WebRTC (desktop client) non-SSO registration behavior

When an alias does not exist in the list of device aliases, a message is written to the support log in the form:

```
Name="support.registration" Message="REST device authentication failed" Reason="Alias not found" Alias="alice@example.com"
```


If the alias exists but the wrong credentials are supplied, a message is written to the support log in the form:

```
Name="support.registration" Message="REST device authentication failed" Reason="Invalid credentials" Alias="alice@example.com"
Expected-username="alice" Supplied-username="slice"
```

Registering and provisioning the Infinity Connect desktop client

The Infinity Connect desktop client can register to a Pexip Infinity Conferencing Node. This enables it to:

- receive calls (as well as place them)
- use directory services to filter and lookup the contact details (phone book) of other devices or VMRs that are set up on the Pexip Infinity platform, making it easier to call those addresses.

 Registration is optional. You do not need to register your device in order to make calls.

The Infinity Connect desktop client can also be provisioned with branding details, allowing it to use the same branding as used by the web app.

The mobile clients do not support provisioning, registration or branding/customization.

This topic covers the client [authentication options](#), the [DNS requirements](#), how to [provision the clients](#), some [example provisioning email template content](#), and a description of the associated [user experience](#).

Client authentication options

When registering an Infinity Connect client to Pexip Infinity, the alias being registered by the client must match one of the entries on the Management Node under **Users & Devices > Device Aliases**. When configuring a device alias, you can specify whether and how an Infinity Connect client that is attempting to register with that alias should authenticate itself (authentication is optional but recommended):

- **SSO:** the client uses Single Sign-On (SSO) services such as AD FS to authenticate the registration.
- **Non-SSO:** the username and password credentials associated with the device alias are used to authenticate the registration.

For any given alias, we recommend that you enable Infinity Connect registrations for either SSO or non-SSO authentication, not both.

Setting up appropriate DNS records

The Infinity Connect desktop client uses its configured **Registration Host** and performs a DNS SRV lookup on `_pexapp._tcp.<registration host address>` to locate a Conferencing Node to which it can send its registration request.

You must therefore ensure that appropriate DNS records have been set up — for more information, see [Setting up DNS records for mobile and desktop client use](#).

Provisioning the Infinity Connect desktop client with registration and/or branding details

Users can manually enter their registration details (alias, credentials, registration host address) into their Infinity Connect desktop client. However, as an administrator you can simplify this process by provisioning individual users with their registration details and automatically applying those registration settings to their Infinity Connect desktop client.

You can also provision the Infinity Connect desktop client with instructions to use the same app branding that has been uploaded to Pexip Infinity (and which is being used automatically by the web app). Note that the mobile clients do not support provisioning, registration or branding.

You perform these provisioning tasks by supplying each user with a provisioning URI in the format:

`https://<node_address>/api/client/v2/provision?data=<Base64 encoded name-value pairs>&message=<Base64 encoded message>`

where:

- **<node_address>** is the address of a Conferencing Node. You must ensure that when the end-user attempts to provision their client that they are able to reach the specified node.
- **<Base64 encoded name-value pairs>** are the data values used to provision the client, and are described below.
- **<Base64-encoded message>** is the provisioning message that is displayed to the user. The **message** parameter is optional and by default is "Your Pexip App should have opened and asked to be provisioned. You can now close this window."

Base64 encoding is used to ensure that the data does not get modified by email clients. Note that Base64-encoded data is not encrypted.

For example, the provisioning URI might look like this:

`https://px01.vc.example.com/api/client/v2/provision?data=ZzUmVkaXJl...etc...D%3D&message=bkgY3VzdG9tIG1lc3Nh`

This provisioning URI can be inserted into email messages without the risk of the link being disabled (unlike the [alternative pexip-provision://](#) URI scheme). This means users will have a directly clickable link without needing to copy and paste the link into their web browser.

Provisioning name-value pairs

The name-value pairs that can be provisioned in the **data** query string parameter are described in the following table. If you use Pexip Infinity to bulk provision device aliases and generate emails to each user, you can use the provided template variables and custom Pexip filters to obtain the values for some of the data items and generate the relevant URIs for each user/client.

Each name-value pair must be separated by an **&**. For example (prior to Base-64 encoding):

`name=Alice®istrationHost=px01.vc.example.com®istrationAlias=alice@example.com®istrationUsername=alice®istrationPassword=password123`

The table shows the common data items, and the additional data items that are used for AD FS SSO authentication:

Name	Value	Suggested sync template variable
name	The name of the user as it will appear to other conference participants.	device_username
registrationHost	The domain, IP address or FQDN of the Conferencing Node to which the client should register, for example <code>px01.vc.example.com</code> . For more information, see Setting up DNS records for mobile and desktop client use .	There is no suitable variable for this, as it is not a user specific value.
registrationAlias	The alias of the device to register to Pexip Infinity.	device_alias

Name	Value	Suggested sync template variable
registrationUsername	The username associated with the device alias (registrationAlias). This does not apply if you are using SSO services.	device_username
registrationPassword	The password associated with the device alias (registrationAlias). This does not apply if you are using SSO services.	device_password
brandingURL	A reference to a directory (on an accessible server) that contains customized branding configuration. You typically use this to instruct the desktop client to use the same branding as the web app. Prior to version 1.8 of the desktop client, the branding package could be hosted on a Conferencing Node. From 1.8 this is not allowed and the package must be hosted on a different external server. See Customizing the Infinity Connect clients for more information.	There is no suitable variable for this, as it is not a user specific value.

Additional data items when using AD FS SSO authentication

adfsFederationServiceName †	The Federation Service name e.g. adfs.example.com.	There are no suitable variables for these items, as they are not user specific values.
adfsResource †	The Resource Identifier e.g. https://pexipappsso.local.	
adfsClientID †	The Client ID e.g. a2a07b42-66d7-41e4-9461-9d343c25b7f3.	
adfsRedirectURI	<p>This is the URI you want the user to be redirected back to after they sign into AD FS. It does not correspond with a value configured on the Management Node but it must be one of the redirect URIs you set up when configuring AD FS on your Windows Server. We recommend you use:</p> <p><code>https://<address>/api/client/v2/oauth2_redirect</code> where <address> is the FQDN of a Conferencing Node or reverse proxy, for example <code>https://px01.vc.example.com/api/client/v2/oauth2_redirect</code>.</p> <p>When the <code>oauth2_redirect</code> page loads it opens the Infinity Connect client to complete the sign-in process. The <code>oauth2_redirect</code> page will remain open but it displays a message which by default is "You have successfully signed in. You can now close this window."</p> <p>You can change this message by including the optional base64-encoded message parameter on the <code>oauth2_redirect</code> page URL. For example, the message "my custom message" is "bXkgY3VzdG9tIG1lc3NhZ2U=" when base64-encoded. You would then specify the <code>adfsRedirectURI</code> as follows:</p> <p><code>https://confnode.example.com/api/client/v2/oauth2_redirect?message=bXkgY3VzdG9tIG1lc3NhZ2U=</code></p>	

† These AD FS related data values should correspond to what you have configured in Pexip Infinity (Users & Devices > AD FS Authentication Clients) for the OAuth 2.0 Client.

Notes:

- You do not have to provision all of the common name-value data items — if you supply a subset of the data, the user can manually enter the additional data if required.
- When using AD FS SSO provisioning, all of the AD FS data items must be included in the provisioning data.

Example device email template content

The following example content for a device provisioning email template shows how you can build the relevant URI with base64-encoded provisioning data (using device provisioning variables populated from LDAP) and provide a clickable link for the recipient of

the email that will provision their client. The first line in this example defines and sets various variables and the second line incorporates those variables in the paragraph text and link that is displayed to the recipient.

```
{%set provisiondata = "name=" + device_username|capitalize +
"&registrationHost=confnode.example.com&registrationAlias=" + device_alias +
"&registrationUsername=" + device_username + "&registrationPassword=" + device_password
%}

<p>You can open <a href="https://confnode.example.com/api/client/v2/provision?{{pex_url_
encode(('data', provisiondata|pex_base64))}}">this link</a> to automatically configure
your client.</p>
```

Remember to substitute **confnode.example.com** with the address of your Conferencing Node.

You can extend the previous example and include the **message** URL parameter (set to 'Provision your app' in this example) in the provisioning link (the **%set** statement is identical to the previous example):

```
{%set provisiondata = "name=" + device_username|capitalize +
"&registrationHost=confnode.example.com&registrationAlias=" + device_alias +
"&registrationUsername=" + device_username + "&registrationPassword=" + device_password
%}

<p>You can open <a href="https://confnode.example.com/api/client/v2/provision?{{pex_url_
encode(('data', provisiondata|pex_base64), ('message', 'Provision your app'|pex_
base64))}}">this link</a> to automatically configure your client.</p>
```

AD FS SSO examples

This is an example of a provisioning link which can be used to set up Single Sign-On via AD FS:

```
{%set provisiondata = "name=" + device_username|capitalize +
"&registrationHost=confnode.example.com&registrationAlias=" + device_alias +
"&adfsFederationServiceName=adfs.example.com&adfsResource=https://pexipappsso.local&adfs
ClientID=a2a07b42-66d7-41e4-9461-
9d343c25b7f3&adfsRedirectURI=https://confnode.example.com/api/client/v2/oauth2_redirect"
%}

<p>Simply open <a href="https://confnode.example.com/api/client/v2/provision?{{pex_url_
encode(('data', provisiondata|pex_base64))}}">this link</a> to configure your client
automatically.</p>
```

Remember to substitute **confnode.example.com** with the address of your Conferencing Node, and to set the **adfsFederationServiceName**, **adfsResource** and **adfsClientID** variables with the appropriate values for your AD FS service.

This next example shows how to include the "successfully signed in" message URL parameter (set to 'Successfully signed-in message' in this example) in the **oauth2_redirect** link:

```
{%set provisiondata = "name=" + device_username|capitalize +
"&registrationHost=confnode.example.com&registrationAlias=" + device_alias +
"&adfsFederationServiceName=adfs.example.com&adfsResource=https://pexipappsso.local&adfs
ClientID=a2a07b42-66d7-41e4-9461-
9d343c25b7f3&adfsRedirectURI=https://confnode.example.com/api/client/v2/oauth2_
redirect?" + pex_url_encode(('message', 'Successfully signed-in message'|pex_base64)) %}

<p>Simply open <a href="https://confnode.example.com/api/client/v2/provision?{{pex_url_
encode(('data', provisiondata|pex_base64))}}">this link</a> to configure your client
automatically.</p>
```

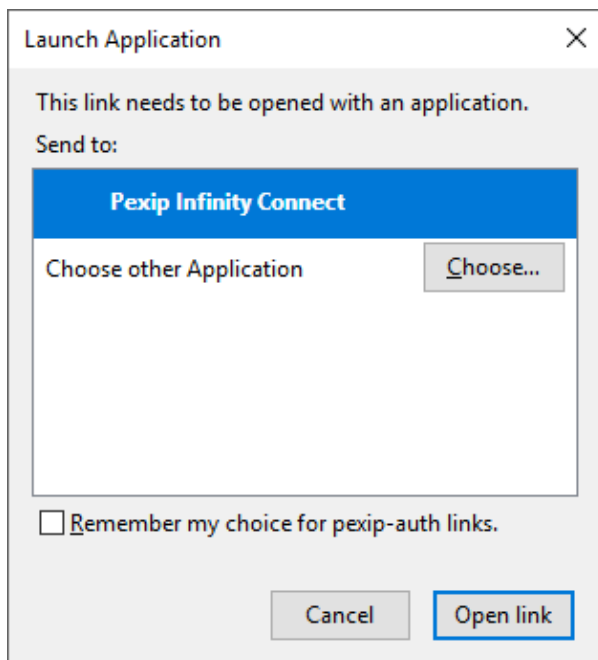
This final example shows how the "successfully signed in" message (on the **oauth2_redirect** URL) and the "provision your app" message (on the **provision** URL) can be customized:

```
{%set provisiondata = "name=" + device_username|capitalize +  
"&registrationHost=confnode.example.com&registrationAlias=" + device_alias +  
"&adfsFederationServiceName=adfs.example.com&adfsResource=https://pexipappsso.local&adfs  
ClientID=a2a07b42-66d7-41e4-9461-  
9d343c25b7f3&adfsRedirectURI=https://confnode.example.com/api/client/v2/oauth2_  
redirect?" + pex_url_encode(('message', 'Successfully signed-in message'|pex_base64)) %}  
  
<p>You can open <a href="https://confnode.example.com/api/client/v2/provision?{{pex_url_  
encode(('data', provisiondata|pex_base64), ('message', 'Provision your app'|pex_  
base64)}}">this link</a> to automatically configure your client.</p>
```

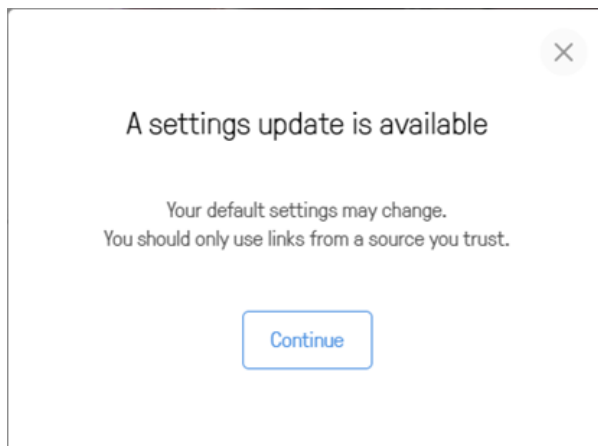
User experience when using the provisioning link

Non-SSO provisioning

When the user clicks on the provisioning link, they are typically asked to confirm or authorize the launch of the Infinity Connect application (the exact nature of the request varies according to the platform and the method of launching the link) and then the Infinity Connect client will launch and present the user with a confirmation screen:



1. Select **Open Link** to launch Infinity Connect.

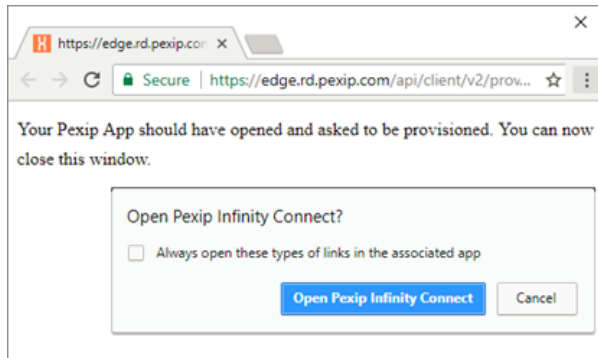


2. Select **Continue** to apply and save the settings contained in the provisioning link.

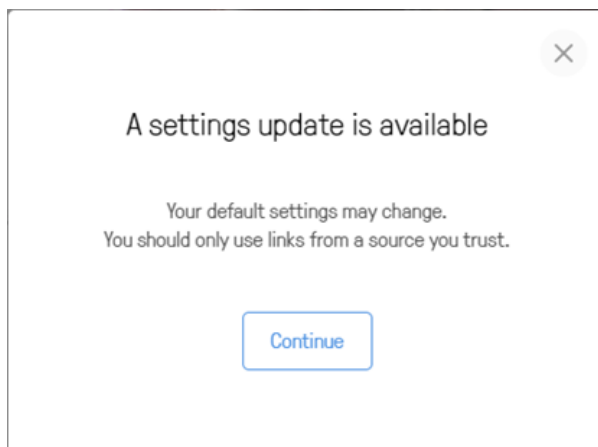
The registration settings in the client are read-only when the client is successfully registered — you must **Unregister** if you want to change them.

AD FS SSO provisioning

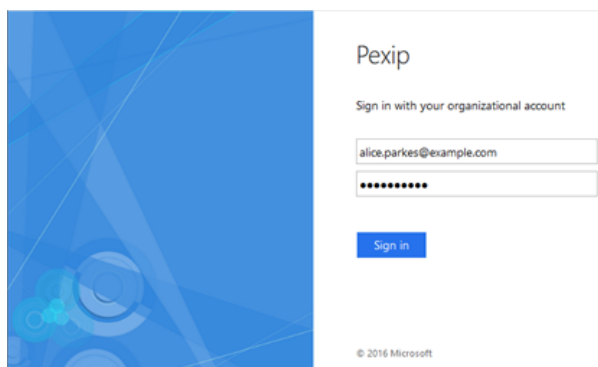
When AD FS SSO provisioning is used, the user is also prompted to sign in to AD FS with their AD credentials. Here are some examples of the screens that are displayed during the provisioning process (the exact nature varies according to the platform, browser and whether the messages have been customized):



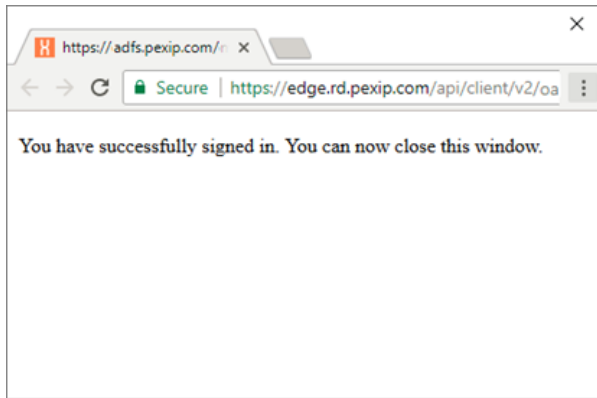
1. Confirm to open the Infinity Connect client.



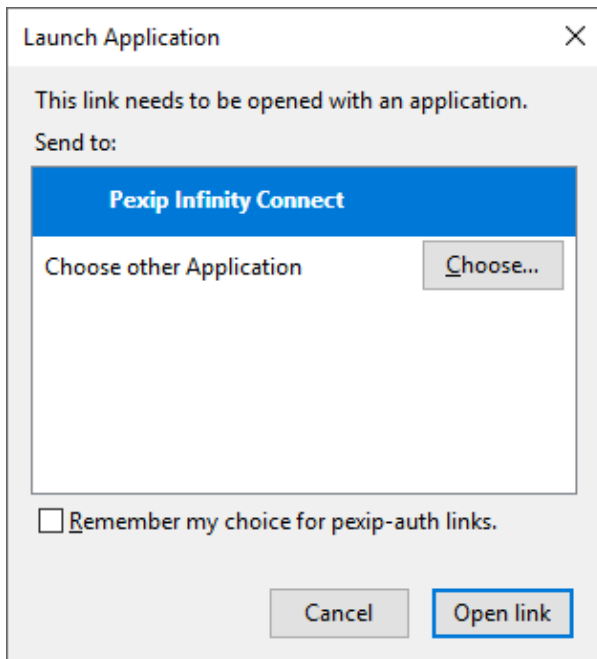
2. Select **Continue** to proceed with provisioning the client.



3. Sign in to AD FS.



4a. AD FS sign-in successful.



4b. Select Open Link to launch Infinity Connect and complete the sign-in process.

When a client has been configured (provisioned) with SSO registration information, the user name / password fields are blank and the registration settings can only be modified by resetting the app.

Alternative pexip-provision:// URI provisioning scheme

When the Infinity Connect desktop client installs, it registers itself to the **pexip-provision://** URI scheme. This provides an alternative provisioning URI that can be used to configure the client with personalized settings for each user. This URI takes the following format:

pexip-provision://settings/?data=<Base64 encoded name-value pairs>

where **data** is set to the same set of name-value pairs as described above.

- i** We recommend using the **https://<node_address>/api/client/v2/provision** style links instead of the **pexip-provision://** style links, as some mail clients (such as gmail) disable embedded **pexip-provision://** style links and other mail clients (such as Outlook) may present users with a security notice warning that the hyperlink may be unsafe and users must choose to continue in order to launch the application.

The following example content for a device provisioning email template shows how you can build the relevant pexip-provision:// URI with base64-encoded provisioning data (using device provisioning variables populated from LDAP) and provide a clickable link for the recipient of the email that will provision their client.

```
{%set provisiondata = "name=" + device_username|capitalize +
"&registrationHost=px01.vc.example.com&registrationAlias=" + device_alias +
"&registrationUsername=" + device_username + "&registrationPassword=" + device_password
%}

<p>You can open <a href="pexip-provision://settings?data={{provisiondata|pex_base64}}">
this link</a> to automatically configure your client.</p>
```

The generated URI for "this link" will take the form `pexip-provision://settings?data=bmFtZT1...etc...HVhcA==`

Customizing the Infinity Connect clients

The branding and styling of the Infinity Connect clients (web app and desktop) can be customized. This changes the look and feel of the Infinity Connect client regardless of which service is being accessed. (However, the theme-based elements of each individual service may also have been customized — a theme changes the look and feel of the actual conference you have joined, or are trying to join.)

Infinity Connect customization can be used to control:

- default settings such as bandwidth, screen sharing frame rate and so on
- the ability to display an image/logo and accompanying welcome text on a landing page, and to use a custom favicon
- language translations and the default language
- the color scheme for buttons, icons and other graphic indicators; elements can be customized individually or a general color scheme can be applied to all similar items.

To customize the web app you typically create and then upload a branding package to the Management Node. That branding package is then automatically applied to all users of the web app. To apply the same customized branding to the desktop clients you need to use Pexip Infinity's provisioning features to instruct those clients to override their built-in branding and use the customized branding instead.

Branding customizations that are applied to the web app via the Management Node will persist over upgrades to subsequent versions of Pexip Infinity software (although you may need to adapt the customization to cater for any new features when upgrading to a new major release).

The instructions in this topic describe how to [create and upload](#), [edit](#) and [remove](#) a branding package, and how to [apply the branding to the desktop clients](#).

Note that the procedures described here apply a generic customization for all Infinity Connect users. If you have specific customization requirements, such as hosting multiple different branding customizations for web app users under different URLs on external web servers or reverse proxies, see [Advanced Infinity Connect customization](#).

Creating and uploading a branding package

You must create a branding package before you can upload it to the Management Node or use it to brand the desktop client. The recommended method to create a branding package for the Infinity Connect clients is to use the Pexip branding portal (<https://brandingportal.pexip.com>).


Creating a branding package via the Pexip branding portal

You can use the Pexip branding portal to customize the Infinity Connect web and desktop clients. This web-based tool guides you through the selection of your image files and colors without having to edit individual CSS files etc, and then generates the customized branding package for you.

To use the Pexip branding portal to generate your branding package for the Infinity Connect clients:

1. Go to the Pexip branding portal (<https://brandingportal.pexip.com>), select the **Next-generation** platform and sign in.
First time users need to register before they can use the portal.
You can also use the portal to create customizations for the legacy web app if required.
2. Select which version of Pexip Infinity you have installed, so that the relevant branding and customization features can be offered.

3. From here you can choose to create new customizations, or edit an existing customization that you have previously created. Configure your customization as required, selecting the relevant image files, colors and settings:
 - The **App Editor** changes the look and feel of the Infinity Connect clients, including enabling an image/logo on the landing page.
 - The **Customizations** section controls the client's configuration settings, including default options, languages and plugins.
 - The **Splash Screens** section doesn't directly affect the Infinity Connect clients. It is used to customize the Pexip Infinity themes (which are used when you join a VMR or other service either via an Infinity Connect client or other endpoint) and generates a separate ZIP package when built.
 - The **Languages** section allows you to set up additional languages for the Infinity Connect clients, or to create a modified version of the default English text strings. When creating a new set of language strings the **Name** is the name you will see within the portal, and the **Label** is the name users will see within the app; the **Locale** enables that language to be used automatically if it matches the browser's default language. If you set up new language option then you must use the **Customizations** section to select the new/modified languages you want to include in your branding package (and deselect the original English language strings if required).
 4. When you have finished configuring your branding, go to the Dashboard, select the relevant **App Edits** and **Customizations** and then **Build** your customization package. If you have added new languages they are automatically included in your build depending upon which languages are selected in the Customization.
This creates and downloads a **branding.zip** file containing your client customizations.
 5. Upload the branding package to your Management Node:
 - a. Go to **Services > Web App Customization**.
 - b. In the **Upload Web App branding** section, select **Choose File** and select the ZIP file containing your customizations.
 - c. Select **Upload branding**.
The branding package will be uploaded. The upload process automatically detects which type of app branding is contained in the ZIP file and processes it accordingly.



Wait for the new branding to be replicated out to all Conferencing Nodes (typically after approximately one minute).
-  This branding package is used to customize the web app by default, but you can also automatically [apply the same branding to the desktop clients](#).

Manually configuring the branding files

Manual configuration is useful if you have plugins or very specific modifications that you want to apply to the branding files. Note that manual configuration requires knowledge of core web-design technologies such as HTML, JavaScript and CSS.

To manually configure the branding files:

1. Download the default web app branding files from the Management Node:
 - a. Go to **Services > Web App Customization**.
 - b. Select **Download** (next to the **Download default branding** label). This downloads a **branding_nextgen_and_legacy_default.zip** file to your local file system.

Note that if you have existing branding files uploaded, you can choose to download those instead of the default files. You can also use files that were originally created by the Pexip branding portal — both methods use the same set of configuration files — you can use the branding portal to apply your basic customization requirements and then make further manual amendments to the configuration files if necessary.
2. Unpack the downloaded file and apply your modifications to the relevant files.
3. Repackage your branding files into a single ZIP file (<name>.zip).
 -  The ZIP file does not have to contain the complete set of branding files. You can upload a subset of the branding files, but you must retain the original file/folder structure in the rebuilt ZIP file. For example, if you have no need to customize the legacy web app files you only need to zip up the **webapp2** folder.
 -  You must include the **manifest.json** file in the **webapp2** folder.

If you are customizing the legacy web app and want to change the **background.jpg** or **logo.png** graphics files, you must also include a **brand.css** file that at least includes the references (**brand-logo** and **brand-background** classes) to those customized images.

4. Upload the branding package to your Management Node:
 - a. Go to **Services > Web App Customization**.
 - b. In the **Upload Web App branding** section, select **Choose File** and select the ZIP file containing your customizations.
 - c. Select **Upload branding**.

The branding package will be uploaded. The upload process automatically detects which type of app branding is contained in the ZIP file and processes it accordingly.

Wait for the new branding to be replicated out to all Conferencing Nodes (typically after approximately one minute).

You can now test the branding by dialing in to one of your Pexip Infinity services via the Infinity Connect web app.

Editing an existing branding package

You can modify an existing branding package by either returning to the Pexip branding portal, or manually editing the branding files that were uploaded previously to the Management Node.

Note that when you upload a new branding package to the Management Node all of the previous branding files for that app are deleted and replaced with the new set of files.

Using the branding portal

If you initially created your branding package via the Pexip branding portal, you can return to the portal and change those files:

1. Go to the Pexip branding portal (<https://brandingportal.pexip.com>) and sign in.
2. Make your changes, previewing them if necessary, and then download a new ZIP file.
3. On the Management Node, go to **Services > Web App Customization** and upload your new branding ZIP file.

Wait for the new branding to be replicated out to all Conferencing Nodes (typically after approximately one minute).

Manually changing your existing web app branding on the Management Node

You can manually edit the existing branding files that have been uploaded to the Management Node (even if those files were originally created via the Pexip branding portal):

1. On the Management Node, go to **Services > Web App Customization**.
2. Download the existing branding files:
 - Select the **Download** option next to the **Download default branding** label to download the system default branding files.
 - If customized branding has been uploaded, you can download it by selecting the **Download** option next to the **Download current branding** label.

(Or the **Download** option next to the **Download legacy branding** label for customized legacy clients.)

3. Unpack the downloaded file and apply your modifications to the relevant files.
4. Repackage your modified branding files into a new ZIP file.

If you are also modifying the legacy web app files, you can package the legacy and webapp2 files as two separate ZIP files i.e. one ZIP containing legacy branding and one ZIP containing webapp2 branding, thus matching the ZIP packages you downloaded. You can also combine them into one ZIP package, but it must match the file structure that is produced when downloading the default branding files.

5. Upload the new ZIP file back onto the Management Node (**Services > Web App Customization** then **Choose File** followed by **Upload branding**).

The upload process automatically detects which type of app branding is contained in the ZIP file and processes it accordingly.

Wait for the new branding to be replicated out to all Conferencing Nodes (typically after approximately one minute).

Removing a web app branding package (revert to default branding)

If you want to revert to the default branding for the web app, you need to remove your customized branding from the Management Node. To do this:

1. On the Management Node, go to **Services > Web App Customization**.
2. From the bottom-right corner of the page, select **Remove branding** to remove any branding (or **Remove legacy branding** if customized branding has been uploaded for the legacy clients).

Wait for the customized branding to be removed from all Conferencing Nodes and for the web app to revert to the default branding (typically after approximately one minute).

Applying branding to the desktop clients

Any branding package that is uploaded to the Management Node is only applied to the Infinity Connect web app.

To apply the same customized branding to the desktop clients you need to use Pexip Infinity's provisioning features to instruct those clients to override their built-in branding and use the customized branding instead. This is achieved by specifying the **brandingURL** provisioning parameter when you construct each individual desktop client user's provisioning URI.

- The **brandingURL** parameter must refer to a directory on an accessible server that contains the branding package.
- The branding package must be signed, and the client must upload a trusted (public) key before the branding can be applied.
- The branding package must be presented as a **branding.zip** file and an associated **branding.zip.sig** file.

For example, if **brandingURL** = **pexample.com/foo**, then you need to provide **pexample.com/foo/branding.zip** and **pexample.com/foo/branding.zip.sig**.

After an Infinity Connect client has been provisioned with a **brandingURL** provisioning parameter, every time it launches it checks the contents of the branding files at the brandingURL location to see if the branding has changed (it checks to see if the **brandingID** in the **manifest.json** file has changed). If the branding has been updated, the client fetches and caches the relevant files.

Note that the desktop client's favicon, taskbar/tray icons and app name cannot be updated via branding as these elements are fixed during the installation of the client software.

See [Registering and provisioning the Infinity Connect desktop client](#) for full instructions about how to set up provisioning URIs. Note that the client does not need to be registered in order to use the branding provisioning feature.

Note that as of version 1.8 you cannot apply branding to the mobile clients, and the desktop client branding can no longer be hosted on Conferencing Nodes.

Creating and signing a branding package for the desktop clients

The branding package in the **brandingURL** location must be presented as a **branding.zip** file plus an associated **branding.zip.sig** file that contains the package's signature.

Contents of branding.zip

Typically we recommend that you use a **branding.zip** file produced by the Pexip branding portal as this is a suitable zip file/format and contains all of the relevant content (although you must still sign it yourself).

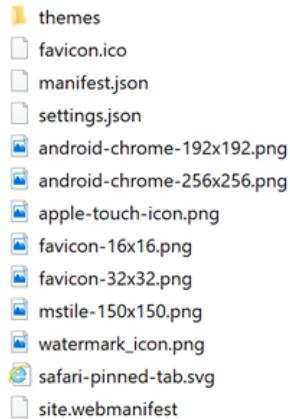
The **manifest.json** is automatically generated by the Pexip branding portal and includes the **brandingID** timestamp and also indicates which parts of the app are customized.

If you want to create your own **branding.zip** file then it must contain a **webapp2** folder as its root folder and that must then have the following structure/contents:

- **manifest.json** (mandatory)
- **settings.json** (optional)
- **watermark_icon.png** (optional)
- **favicon** files (optional, applies only to the web app)
- **site.webmanifest** (optional)
- **themes** directory containing **styles.css** (both optional)

as shown below:

Name



Full details of the structure of the manifest.json file and the other application files are contained in [Advanced Infinity Connect customization](#).

Signing the branding package

You must use JSON Web Token (JWT) to sign the package. (JWT is an open standard that defines a way for securely transmitting information between parties as a JSON object.)

As part of the process to sign the branding package you need a public/private keypair. You may already have a keypair that you can use for this process, or you can use a third-party tool such as PuTTYgen to generate a keypair. The key must be in RSA format and at least 2048 bits.

To sign the branding package and create your .sig file:

1. Create your **branding.zip** file.
2. Using a plain text editor, create a shell script file called **mkjwt.sh** containing the following code:

```
#!/bin/sh

set -e

if [ $# -ne 2 ]; then
    echo "Usage: $0 <privatekey.pem> <branding.zip>" >&2
    exit 1
fi

HEADER="eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9"
HASH=$(openssl dgst -sha256 $2 | sed -e 's/^[^ ]* //' )
PAYLOAD=$(echo -n "{\"sha256\":\"${HASH}\"}" | base64 -w0 | sed -e 's/\+/-/g' -e 's/_/_/g' -e 's/=//g')
SIG=$(echo -n "${HEADER}.${PAYLOAD}" | openssl dgst -sha256 -sign $1 | base64 -w0 | sed -e 's/\+/-/g' -e 's/_/_/g' -e 's/=//g')

echo "${HEADER}.${PAYLOAD}.${SIG}"
```

3. Copy your private key file (named **privatekey.pem**), the **branding.zip** file and the **mkjwt.sh** file into the **/dev/shm** directory on the Management Node using an SCP (Secure Copy) client, for example WinSCP.
4. Connect over ssh into the Management Node as user admin with the appropriate password.
5. Run the following commands:


```
cd /dev/shm
chmod 0755 mkjwt
```

6. Run the following command to generate the .sig file:
`./mkjwt privatekey.pem branding.zip >branding.zip.sig`
7. Run the following command to remove the private key file:
`rm ./privatekey.pem`
8. Use the SCP client to copy the generated **branding.zip.sig** file to your local machine.

Using the branding package on the desktop client

The client will not automatically use the customized branding package (as referred to at the provisioned **brandingURL** location).

Each client user must first import a trusted key via **Settings > Advanced settings > Import trusted key** and confirm that they want to apply the branding. The trusted key file they need to import (i.e. that you need to distribute) is the public key file used as part of the key pair used to create the JWT signature.

 Only distribute the public key. Do not distribute the private key.

Pexip Infinity conference settings

About aliases and access numbers

Every Virtual Meeting Room (VMR), Virtual Auditorium, Virtual Reception, scheduled conference, and Test Call Service has one or more aliases associated with it.


When Pexip Infinity receives an incoming call via one of its Conferencing Nodes, it checks whether the destination alias belongs to a Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service; if so, it will direct the call to that service. If the alias does not belong to any of the above services, Pexip Infinity will then check through the Call Routing Rules used by the [Infinity Gateway](#) to see if the alias matches any rules specified there - for more information, see [Service precedence](#).

When Pexip Infinity receives a call to a **Virtual Meeting Room** (including scheduled conferences) or **Virtual Auditorium** alias, it creates a conference instance and routes the call to that conference. Any further calls received to any of the aliases belonging to the same Virtual Meeting Room or Virtual Auditorium are routed to the same conference instance, for the duration of that particular conference. If the service has more than one alias, participants can dial any one of the aliases and be routed to the same conference instance. For more information, see [Using multiple aliases to access the same service](#).

Virtual Receptions also have one or more aliases associated with them. In these cases, when the participant dials a Virtual Reception alias they will be taken to the Virtual Reception IVR service, from which they can use a DTMF keypad to enter the number of the Virtual Meeting Room or Virtual Auditorium they wish to join. This number must correspond to a numeric-only alias of the Virtual Meeting Room or Virtual Auditorium in question. For more information, see [About the Virtual Reception IVR service](#).

Pexip Infinity's **Test Call Service** provide a test loopback service that allows users to check the quality of their video and audio, and verifies that they can connect to a Conferencing Node. See [Configuring the Test Call Service](#) for more information.

In most cases, the alias received by Pexip Infinity will be the same as the alias that the conference participant dialed from their endpoint, but there are some exceptions, described in [Search rules and alias transforms](#) and [ENUM](#).

 For simplicity, the following discussion assumes that the alias dialed by the endpoint user is the same as the alias received by Pexip Infinity.

Creating, editing and viewing aliases

Each alias is associated with a single service, but a service can have more than one alias.

To view a list of all existing aliases for all services, go to **Services > Aliases**. From here you can:

- Sort the list of aliases alphabetically by **Alias** or by **Service** name.
- Search for a particular alias.
- Add a new alias.
- Modify an alias, including changing the service with which it is associated.

You can also add or modify aliases while creating or editing a service.

When adding or editing aliases, the options are:

Option	Description
Service	Select the name of the Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service to access when participants dial this alias.
Alias	<p>The alias that, when received by Pexip Infinity, will cause it to route the call to this service (Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service).</p> <p>The alias entered here must match the alias as it is received by Pexip Infinity. Wildcards and regular expressions are not supported.</p> <p>You may also want to define multiple aliases for the same service to ensure that it can be accessed by devices and protocols that enforce specific alias formats — for more information, see Using multiple aliases to access the same service.</p>
Description	An optional description of the alias. This is useful if you have more than one alias for a service. Note that this description may be displayed to end users on registered Infinity Connect clients who are performing a directory search.

Restrictions

An alias can include a domain or subdomain but this is optional (see [Domains](#)). Aliases can contain numbers, letters, dashes and dots. In certain circumstances an alias can also be in the form of an IP address — for more information see [Dialing by IP address](#).

Aliases do not support wildcards or regular expressions.

Case insensitivity

The aliases that you configure on Pexip Infinity are not case-sensitive, and Pexip Infinity treats all incoming aliases as not case-sensitive. For example, this means that:

- a user who dials `meet.alice` will match against a VMR with an alias of `Meet.Alice`
- a user who dials `Meet.Alice` will match against a VMR with an alias of `meet.alice`

Ignoring IP addresses

Pexip Infinity ignores any IP address or IP address and port combination appended to an alias. This is because some SIP endpoints automatically add the IP address of their proxy to the URI that is dialed by the user.

For example, a VMR with a single alias of `meet.alice` will be matched when an endpoint dials any of:

- `meet.alice`
- `meet.alice@<IPAddress>`
- `meet.alice@<IPAddress>:<port>`

Dialing by IP address

We do not recommend dialing by IP address. However, to support endpoints that can only dial by IP address, you can give a service an alias that is the same as the IP address of one of your Conferencing Nodes.

In this case, when a user dials the IP address from that endpoint, the call will be routed directly to the Conferencing Node with that IP address. Pexip Infinity will then perform its standard behavior and check to see if the destination alias of the call (which in this scenario will be the dialed IP address) matches any of its configured aliases; if so, the call will be routed to the service (Virtual Reception, Virtual Meeting Room etc.) that is associated with that alias.

If other endpoints, such as Infinity Connect clients, dial the IP address alias, they will also be connected to the appropriate service (although the Conferencing Node they connect to is determined via other means).

In most cases you would use this feature to assign a Conferencing Node IP address as an alias for a Virtual Reception, so that users could then select which Virtual Meeting Room or Virtual Auditorium they wish to join. For more information, see [About the Virtual Reception IVR service](#).

Ignoring protocol prefixes

Pexip Infinity ignores relevant URI schemes included in an alias, as in the case where a SIP endpoint automatically prefixes the URI dialed by the user with "sip:". The ignored prefixes are "sip:", "sips:" and "h323:". This is because the protocol used for the call does not matter to Pexip Infinity.

For example, a VMR with a single alias of `meet.alice` will be matched when an endpoint dials any of:

- `sip:meet.alice@IPAddress`
- `sips:meet.alice@IPAddress`
- `h323:meet.alice`

Search rules and alias transforms

Some call control systems (for example the Cisco VCS) support the use of search rules and alias transforms. In these cases, the alias that was dialed by the endpoint user and received by the call control system may be changed by the call control system in some way before it is passed on to Pexip Infinity. Often this is done in larger deployments to support complex dial plans.

For example, the call control system might have a rule that replaces the domain `example.com` with `example.net`. This means that when a user dials `meet.alice@example.com`, the call is routed to `meet.alice@example.net`. Therefore it is this latter alias that must be configured on Pexip Infinity in order to match it with a VMR.

ENUM

The ENUM system allows users to dial an E.164 number (for example a telephone number) which is then mapped using DNS to a SIP URI. For more information, see [RFC 3761](#).

For example, your dial plan could be set up so that when a user dials `555 0123`, the call is routed via ENUM to `meet.alice`.

If your dial plan uses ENUM, the resulting SIP URIs must be included as aliases on Pexip Infinity in order to match them with a VMR.

Domains

Domains are optional in aliases. However, when a call is received to an alias in the form of a URI that includes a domain, the domain is not ignored.

For example, if a VMR is configured with an alias of `meet.alice`, users can dial `meet.alice` or `meet.alice<IPAddress>` to access the meeting room. However, if they dial `meet.alice@example.com` they will not be able to access the VMR because the VMR alias does not include the domain.

When a SIP endpoint user dials an alias that does not include a domain (for example `meet.alice`), the SIP endpoint will automatically add its own domain to the alias (making it for example `meet.alice@example.com`). So even if the VMR is configured with an alias of `meet.alice` and this alias is dialed by a participant from a SIP endpoint, the participant will not be able to join the conference. (H.323 endpoints do not add domains.)

In the absence of a call control system to strip the domain part of the alias, you could instead add to the VMR a second alias of `meet.alice@example.com` so that participants can dial `meet.alice` from either a SIP or H.323 endpoint and access the same conference.

When integrating Pexip Infinity with an on-premises Microsoft Skype for Business / Lync environment, where the main SfB/Lync domain is `example.com`, you may want to use a subdomain such as `vc.example.com` for the Pexip Infinity / VTC systems. This can simplify the routing rules between systems as it allows SfB/Lync users and SfB/Lync meetings to be addressed via `user@example.com` or `conference_id@example.com` respectively, while calls to VTC systems can be placed via `device@vc.example.com`.

Using multiple aliases to access the same service

You may want conference participants to be able to dial different numbers to access the same [Virtual Reception](#), or to allow different conference participants to dial different aliases but still end up in the same VMR.


Thus, depending on your deployment environment and the types of devices and protocols used to dial into your services, you may want to configure a variety of alias formats to allow access to those services, including:

- **name@domain** (for full URI dialing such as from SIP and Skype for Business / Lync clients)
- **name** (for systems that strip off or do not require a domain)
- **numeric/E.164** (for endpoints that do not support URI dialing, or can only dial via IP address, or for audio-only callers via PSTN/ISDN gateways that need to be routed via a Virtual Reception).

It is often good practice to create a URI and a numeric alias as a minimum. If numeric/E.164 formatted aliases are used as your primary dial plan, you could also reuse the same number in a `number@domain` URI-style alias. Some [example alias dial plans](#) are given below.

You can add any number of aliases to a single service (Virtual Meeting Room, Virtual Reception etc). Note that the order in which aliases appear in the list of aliases is relevant when [Automatically dialing out to a participant from a conference](#).

Adding additional aliases

To add an additional alias to a service while you are creating it, select  **Add another Alias**.

To add another alias to an existing service:

1. Go to **Services > Virtual Meeting Rooms**, **Services > Virtual Auditoriums**, **Services > Virtual Receptions**, **Services > Scheduled Conferences** or **Services > Test Call Service** as appropriate.
2. Select the name of the service you wish to add the alias to.

3. In the **Aliases** section at the bottom of the page, enter the new alias in the empty **Alias** field.
4. Select **Save**.

If you have [provisioned your VMRs](#) from Active Directory via LDAP and you want to manually change or add aliases to those VMRs, you may want to ensure that **Allow aliases to be overridden** is enabled on the LDAP sync templates used to generate those VMRs, otherwise any changes you make will be overridden the next time that template is synchronized.

Examples

- If some of your endpoints do not support URI dialing, you could set up a VMR for Alice with one alias of **meet.alice.jones** and another alias of **555 25423**. This would give conference participants two different methods of accessing the same conference.
- If your deployment includes a Virtual Reception, every Virtual Meeting Room and Virtual Auditorium that you want to be accessible from the Virtual Reception needs to have a numeric-only alias (so that participants can enter the number using DTMF), in addition to any aliases in the form of a URI. So to continue with our example, you could add a third alias to Alice's VMR of **25423**, which is the number that participants would enter from the Virtual Reception. For more information, see [About the Virtual Reception IVR service](#).
- If you use an ISDN gateway to support telephone participants, you could set up Alice's VMR with aliases of **meet.alice.jones** and an appropriate E.164 number.
- If you do not have or do not wish to use a call control system to transform aliases, you could set up Alice's VMR with aliases of **meet.alice.jones** and **meet.alice.jones@example.com** to support internal and external dialing, and dialing from both SIP and H.323 endpoints.

About PINs, Hosts and Guests

For added security, you can set up your Pexip Infinity Virtual Meeting Rooms (including scheduled conferences) and Virtual Auditoriums with PIN numbers. You can use the same PIN for all participants, or you can use PINs to differentiate between Hosts and Guests.

When a participant dials the alias of a conference that has a PIN, they are presented with an Interactive Voice Response (IVR) screen where they are asked to enter the PIN number. They must enter the correct PIN before they can join the conference. This topic covers:

- [Using the same PIN for all participants](#)
- [Using PINs to differentiate between Hosts and Guests](#)
- [Creating a Hosted Virtual Meeting Room or Virtual Auditorium](#)
- [How to combine Host PINs and Guest PINs for differing levels of security](#)
- [Allowing end-users to manage their PINs](#)
- [Incorrect PINs](#)
- [Using # at the end of a PIN](#)
- [Changing a participant's role from Guest to Host \(and vice versa\)](#)
- [Including the PIN in the dial string to bypass the PIN entry screen](#)
- [Limiting the time a participant can spend at the PIN entry screen](#)
- [Limiting how long Guests can wait for a Host](#)

Using the same PIN for all participants

All participants in a conference that uses a single PIN have the same [Host privileges](#). (To create a PIN-protected conference where participants have different levels of privileges, see [Using PINs to differentiate between Hosts and Guests](#).)

To set a PIN number on a conference:

1. Go to **Services > Virtual Meeting Rooms**, **Services > Virtual Auditoriums** or **Services > Scheduled Conferences** and select the service you wish to change.
2. In the **Host PIN** field, enter the PIN number to be used.
3. In the **Allow Guests** drop-down list, select **No**.

Note that:

- PINs must use the digits 0-9 only.
- PINs may optionally end with #.

- PINs must be between 4–20 digits long, including any #.

Using PINs to differentiate between Hosts and Guests

You can set up your Virtual Meeting Rooms and Virtual Auditoriums so that they are "Hosted". Hosted conferences have two different types of participants: **Hosts** and **Guests**. Hosts and Guests both dial the same alias to join the conference.

You can create two types of Hosted conference:

- the Host must enter a PIN, but Guests do not require a PIN, or
- both Hosts and Guests must enter a PIN.

In both cases:

- Only those participants who enter the Host PIN have privileges to control the conference; everyone else is a Guest and can only participate in the conference.
- An administrator can configure individual Virtual Meeting Rooms and Virtual Auditoriums so that Guest participants are not allowed to present into the conference (they can still receive presentation content from other Host participants). By default, Guests are allowed to present content.
- If a conference is [locked](#), Hosts can still join, but Guests are held at a waiting screen.
- For conferences that use Virtual Auditoriums, the administrator can select different layouts for the Host view and Guest view. For more information, see [Selecting the layout seen by participants](#).

Guest privileges

- If any Guests join the conference before a Host has arrived, or while the conference is locked, they are shown a holding image and hear a message advising them that they are waiting for the conference Host. The message is repeated every 30 seconds for video participants, and every 15 seconds for audio-only participants. Such participants will be disconnected automatically after 15 minutes waiting, but Administrators can change this time - see [Limiting how long Guests can wait for a Host](#).
- A minute or so after the last Host has left, any remaining Guests are automatically disconnected. Administrators can change the length of time before Guests are disconnected by going to Platform > Global Settings > Service Configuration > Guests-only Timeout.

In addition, if the conference is held in a Virtual Auditorium:

- During the conference, Guests are only able to see the Hosts; they cannot see other Guests. However, after the last Host has left, Guests can see any other remaining Guests during the disconnection timeout period.
- Guest participants will not be shown the streaming indicator and so will not be aware if the conference is being streamed or recorded.

Host privileges

- Guests cannot join a conference until the first Host has joined. However, if the Host joins as a presentation and control-only participant via an Infinity Connect client, that Host has to start the conference manually to allow Guests to join.
- If the first participant to join the conference is a Host, that Host is shown a holding image until another Host or Guest joins the conference.
- If the conference has a Host PIN, participants who enter the Host PIN can join the conference even if it is [locked](#).
- The presence of a Host in a conference may prevent that conference from being [automatically terminated](#).
- A minute or so after the last Host has left, any remaining Guests are automatically disconnected. Administrators can change the length of time before Guests are disconnected by going to Platform > Global Settings > Service Configuration > Guests-only Timeout.
- Hosts can use Infinity Connect clients to:
 - disconnect participants from the conference
 - add participants to the conference
 - mute and unmute an individual participant, or all Guest participants simultaneously
 - lock and unlock a conference, and to allow individual participants into a locked conference
 - change the role of other participants in the conference
 - control another participant's camera (if it supports FECC)

- spotlight a participant in the main video
 - lower the raised hand of another participant
 - send DTMF tones to other participants in the conference.
- The Pexip Infinity administrator may configure a Virtual Auditorium so that Hosts see other participants in a different layout to that seen by Guests. For more information, see [Selecting the layout seen by participants](#).

Creating a Hosted Virtual Meeting Room or Virtual Auditorium

To create a Hosted Virtual Meeting Room or Virtual Auditorium:

1. Go to **Services > Virtual Meeting Rooms**, **Services > Virtual Auditoriums** or **Services > Scheduled Conferences** and select the service you wish to change.
2. In the **Host PIN** field, enter the PIN number to be used by the Hosts.
3. In the **Allow Guests** drop-down list, select **Yes**.
4. If you want to require Guests to enter a PIN, in the **Guest PIN** field, enter the PIN number to be used.
If you leave this field blank, anyone who dials the alias of the Virtual Meeting Room or Virtual Auditorium will be able to access the conference as a Guest. However, the conference will not start until the first Host has joined and in the meantime all Guests will continue to be shown the holding screen.

Note that:

- Host PINs and Guest PINs must be different.
- PINs must use the digits 0-9 only.
- PINs may optionally end with #.
- PINs must be between 4–20 digits long, including any #.
- If the Host PIN ends in # and a Guest PIN is used, the Guest PIN must also end with #.
- If # is not used, Host PINs and Guest PINs must have the same number of digits.
- You cannot configure a Guest PIN unless you have already configured a Host PIN.

How to combine Host PINs and Guest PINs for differing levels of security

The table below shows how to use **Host PINs** (for [Host privileges](#)) and **Guest PINs** (for [Guest privileges](#)) to achieve various levels of security when [creating hosted](#) Virtual Meeting Rooms and Virtual Auditoriums:

Access	Roles	Host PIN	Allow Guests	Guest PIN
Anyone - no PIN required	All participants have the same Host privileges.	Leave blank	Select No	Leave blank
All participants must enter the same PIN	All participants have the same Host privileges.	Enter the PIN	Select No	Leave blank
Hosts must enter a PIN but Guest do not	Participants who enter the PIN have Host privileges. The conference will not begin until they have joined, and will finish a minute or so after the last Host leaves. All other participants have Guest privileges and do not need to enter a PIN. They will see a holding screen until the first Host joins.	Enter the PIN	Select Yes	Leave blank
Hosts and Guests must enter different PINs	Participants who enter the Host PIN have Host privileges. The conference will not begin until they have joined, and will finish a minute or so after the last Host leaves. Participants who enter the Guest PIN have Guest privileges and will see a holding screen until the first Host joins.	Enter the PIN (must be different from the Guest PIN)	Select Yes	Enter the PIN* (must be different from the Host PIN)

* If you configure a **Guest PIN**, you **must** also configure a **Host PIN**.

Allowing end-users to manage their PINs

Host and Guest PIN numbers are initially assigned when the VMR is created by the administrator.

The Pexip VMR self-service portal is a separately-installable component that allows end-users to manage their personal Virtual Meeting Room without having to send requests to their administrator to change the configuration of their VMR.

Incorrect PINs

Participants (other than Infinity Connect clients, and connections via the client API) have three attempts to enter a correct PIN. After the third unsuccessful attempt, they will hear a message advising them that the PIN is invalid, and their call will be disconnected. Further protection against PIN cracking attempts can be achieved using the [Break-in resistance](#) settings, which apply to all clients.

All PIN entry attempts, successful and unsuccessful, are logged in the [administrator log](#).

Using # at the end of a PIN

PINs can optionally end with a # (known either as the **pound** or **hash** character). Using a terminal #:

- Allows administrators to set different length PINs for Hosts and Guests.
- Allows users who have accidentally entered too many or too few digits (particularly those using audio-only, who have no on-screen indication of the number of digits required) to return to the start of the PIN entry process by entering #.
- Adds extra security by disguising the length of the PIN. This is because when a PIN does not end in a #, users (except those using Infinity Connect clients) will be presented with the "Invalid PIN" message after entering the same number of digits as that of the PIN.

You cannot configure a Host PIN that ends with # and a Guest PIN that does not (or vice versa).

For conferences where a terminal # is used, Pexip Infinity automatically uses the appropriate [audio file](#), asking participants to "please enter the conference PIN number followed by the pound key". [Preconfigured themes](#) allow you to change the use of "pound key" to "hash key".

For Infinity Connect clients, the terminal # is optional when entering a PIN.

Examples

- If you want to use a terminal # on a VMR with a Host PIN of 1234 and a Guest PIN of 567890:
 - in the **Host PIN** field enter **1234#**
 - in the **Guest PIN** field enter **567890#**.

Now when anyone calls the VMR they will hear the message "please enter the conference PIN number followed by the pound key".

- If you do not want to use a terminal # on a VMR with a Host PIN of 1234 and no Guest PIN:
 - in the **Host PIN** field enter **1234**
 - leave the **Guest PIN** field blank.

Now when anyone calls the VMR they will hear the message "please enter the conference PIN number".

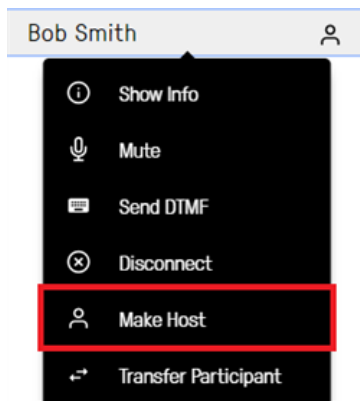
Changing a participant's role from Guest to Host (and vice versa)

Infinity Connect users who have a role of Host can change the role of other participants in the conference. This is useful, for example, if a Host needs to leave the conference; they can promote one of the Guests to a role of Host so that the conference will continue after they disconnect.

You can also use this feature to allow a participant who is waiting at the PIN entry screen to join the conference as Host or Guest without entering a PIN. Such users appear in the roster with a role of **Unknown**. (Note that this does not apply to participants using Infinity Connect clients, as they use a different PIN entry process.)

Participants who have joined via a Skype for Business / Lync meeting will have a role of **External**; their status cannot be changed.

To change a participant's role, from the **Participant list**, select the participant and then select **Make Host** or **Make Guest**:



You cannot change your own role to Guest.

Including the PIN in the dial string to bypass the PIN entry screen

SIP and H.323 endpoints and Skype for Business / Lync clients that dial into PIN-protected conferences can bypass the PIN entry screen by including the PIN in the dialed alias.

This makes it more convenient, for example, when bookmarking PIN-protected conferences.

They can do this by including the relevant PIN (either the Host PIN or the Guest PIN as appropriate) in their dial string when dialing the Virtual Meeting Room or Virtual Auditorium. The dial string should be in the format: `<vmr_alias>**<PIN>@<domain>`.

For example, if the alias of the Virtual Meeting Room is `meet.alice@example.com` and the PIN is `1234`, then the endpoint can dial `meet.alice**1234@example.com` to go straight into the VMR.

Note that H.323 devices can also use the dial format `<vmr_alias>#<PIN>@<domain>`.

If there is a # configured at the end of the PIN e.g. `1234#`, then that # is not required in the dial string.

Limiting the time a participant can spend at the PIN entry screen

By default, participants have two minutes to enter a correct PIN before the system disconnects them. The timer starts when the participant gets to the PIN entry screen, and participants will be disconnected at the end of the specified time, whether or not they have entered any PINs during that time (in other words, the timer does not restart after each PIN entry).

You can change the length of time by going to **Platform > Global Settings > Service Configuration** and changing the **PIN entry timeout**.

Note that this limit does not apply to participants using Infinity Connect clients, or to connections via the client API, because they enter PINs via a different process.

Limiting how long Guests can wait for a Host

By default, a Guest participant will be disconnected if they have been waiting for 15 minutes for a Host to either:

- join the conference
- [allow the Guest to join a locked conference](#), either by unlocking the conference or permitting the individual participant to join.

This time applies per-participant, so if Guest A joins a conference five minutes before Guest B and a Host does not join, Guest A will be disconnected five minutes before Guest B.

You can change this length of time by going to **Platform > Global Settings > Service Configuration** and changing the **Waiting for Host timeout**.

Selecting the layout seen by participants

A layout is the view that each participant has of all the other participants in the meeting. The layout for a meeting is configured in advance by the administrator, but can be changed during the meeting by Host participants using the Infinity Connect client. Some of the options available depend on whether the conference is being held in a [Virtual Meeting Room](#) (including scheduled conferences) or a [Virtual Auditorium](#).

You have a choice between Pexip's AI-driven [Adaptive Composition layout](#) and a range of [alternative layouts](#). By default, the "Large main speaker and up to 7 other participants" (1+7) layout is assigned to each VMR.

Other layout features you can enable include [showing the names](#) of active speakers and participants as text overlays, and [spotlighting](#) (pinning) nominated participants into the primary positions in the stage layout.

Adaptive Composition layout

Adaptive Composition is an intelligent meeting layout with real-time automatic face detection and framing:

- Powered by AI and machine learning to give users a more natural, engaging, video-first meeting experience.
- It continuously analyzes each video feed from all participants and uses automatic face detection and framing to create an optimized view of that participant, or group of participants where there are several people in that video feed.
- Layout placement of each participant is based on a combination of face count (number of faces detected within that participant's video feed) and their recent speaking frequency, so as to allocate more space for larger groups of people.
- Independent of where the participants are and which device they are using. No end-user action or configuration is required.

When using the Adaptive Composition layout:

- A maximum of 12 video participants are shown, spread across one row of 2 large images, a middle row of 3 slightly smaller images and a bottom row of 7 thumbnail images.
- Any inactive video participants are automatically removed from the video layout. This is triggered if the participant:
 - Walks away from the camera, leaving an empty chair in frame i.e. no face is detected.
 - Points the camera at the ceiling or other place where no movement is detected.
 - Physically closes the camera lid or puts a cover over the camera.










Note that these participants still continue to **receive** the full video layout.

The participant is returned to the video layout if video is restored, or if any faces or movement are detected.

- All conference indicators, such as participant counts, audio participants, recording indicators and locked status are shown at the top-center of the layout.
- There is an additional configuration option to show the name of the active speaker, as well as, or instead of, showing the names of all participants — see [Showing the names of active speakers and participants](#) below.

The conference indicators shown at the top-center of the layout include:

	Total participant count.
	Number of participants connected as audio-only.
	Number of inactive video participants and video-muted Infinity Connect participants, who are excluded from the video layout.
	An audio or inactive video participant's display name is shown if they start speaking.
	Conference locked indicator.
	Conference locked/unlocked messages are temporarily displayed when a conference is locked/unlocked.
	Recording / streaming indicator.

This layout increases backplane bandwidth and CPU resource usage.

Note that Adaptive Composition applies face-detection technology, but does not apply any biometric or facial-recognition technology or store any such related data.

Alternative layouts

In addition to Adaptive Composition, you can select from a range of other layouts for your Virtual Meeting Rooms and Virtual Auditoriums:

- Full-screen main speaker only (1 + 0 layout)
- 4 main speakers only (4 + 0 layout)
- Large main speaker and up to 7 other participants (1 + 7 layout) — this is the default layout
- Small main speaker and up to 21 other participants (1 + 21 layout)
- 2 small main speakers and up to 21 other participants (2 + 21 layout)
- 1 small main speaker and up to 33 other participants (1 + 33 layout) — this is a tech preview layout and has to be enabled via **Platform > Global Settings > Tech Preview Features**.

For full details on the differences between these layouts, see [Virtual Meeting Room layouts](#).

Features common to all layouts

The following layout features and characteristics apply to all layout types (Adaptive Composition and non-adaptive layouts):

- Any video-muted Infinity Connect participant is removed from the video mix and is included in the inactive/muted indicator count.
- When sending presentation content to a single-stream endpoint or broadcasting via an RTMP stream (when not dialed out with dual streaming), Pexip Infinity sends the video stream of the active speaker in a small window in the upper right corner of the presentation.
- When a participant is the only device that is sending video, that participant sees a holding screen until other video participants join the conference. The holding screen indicates either that the participant is the only participant in the conference, or that all of the other participants are audio-only (and they are shown via their audio-only indicators as usual). The holding images are fully customizable via the [theme](#) associated with the service (Virtual Meeting Room, Call Routing Rule etc).
- A full list of participants is available using Infinity Connect clients.

Showing the names of active speakers and participants

Each [Virtual Meeting Room](#) and [Virtual Auditorium](#) has the option to show the names of the participants in a text overlay along the bottom of their video image.

To turn on participant names by default for a service:

1. Go to **Services > Virtual Meeting Rooms**, **Services > Virtual Auditoriums** or **Services > Scheduled Conferences** and select the service you want to change.
2. Next to the **Show names of participants** field select **Yes**.
3. For Adaptive Composition layouts you can also choose to **Show name of active speaker**, as well as, or instead of, showing the names of all participants.
 - When participant names and active speaker display are both enabled, the name of the person speaking is shown in green instead of white.
 - When only active speaker display is enabled, the name of the person speaking fades in and then fades out again after a few seconds.

i Host participants using Infinity Connect can enable and disable text overlay while a conference is in progress by using the [commands](#) `/overlay on` and `/overlay off`.

When text overlay is enabled, the name that is shown depends upon whether the participant was invited into the conference, and the type of endpoint device, as follows:

- Automatically dialed participants (regardless of the type of endpoint that was dialed): either the ADP's configured **Participant display name** or, if this field is blank, the **Participant alias**, is shown.
- Manually dialed participants (regardless of the type of endpoint that was dialed): either the participant's configured **Participant display name** or, if this field is blank, the **Participant alias**, is shown.
- Infinity Connect participants: the text that the user entered into the initial **Type your name here** field is shown.

- SIP endpoints: the endpoint's display name is shown.
- H.323 endpoints: the endpoint's alias is shown.
- Skype for Business / Lync clients: the username portion of the user's sign-in address is used, for example if Alice signs in as `alice@example.com` the text `alice` is shown.

The size of the text overlay varies automatically according to the resolution being received by the endpoint and the type of layout. In low resolutions the text overlays are not shown.

The default font for the in-conference display of participant names is Roboto (which cannot be changed), or if that is not available for the character set, Noto Sans.

Allowing end-users to change the default layout

A VMR's settings for its layout and display of participant names are initially assigned when the VMR is created by the administrator.

The Pexip VMR self-service portal is a separately-installable component that allows end-users to manage their personal Virtual Meeting Room without having to send requests to their administrator to change the configuration of their VMR.

Spotlighting a participant (non-adaptive layouts only)

Host participants using an Infinity Connect client can "spotlight" themselves or other participants in the meeting.

The spotlight feature locks any spotlighted participants in the primary positions in the stage [layout](#), ahead of any current speakers. When any participants have been spotlighted, the first one to be spotlighted has the main speaker position, the second one has the second position (leftmost small video, for example), and so on. All remaining participants are arranged by most recent voice activity, as is default.

In a Virtual Auditorium, where Guests cannot see other Guests, if a Guest participant has been spotlighted they will only appear in the main video for Host participants; other Guests will not be able to see the spotlighted Guest.

When [Lock presenter as main speaker](#) has been selected, a spotlighted participant will appear in the main video ahead of any presenters.



To spotlight a participant, from the Infinity Connect client **Participant list**, select the participant and then select **Spotlight**.

i Host participants using Infinity Connect can also use the [commands](#) `/spotlight on [participant]` and `/spotlight off [participant]`.

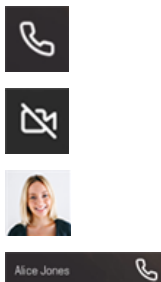
Endpoints that support FECC can also be spotlighted [while the FECC dialog is open](#).

Layout features common to all non-adaptive layouts

All of Pexip Infinity's non-Adaptive Composition layout types provide similar features to Adaptive Composition in terms of showing participant numbers, lock status and so on, but these various indicators are in different positions within the layout:

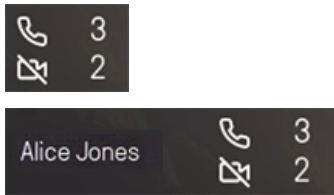
- When there are between one and four audio-only or inactive-video participants, each is represented by an indicator on the left side of the video window. By default this is either the audio-only icon  or the video-muted icon . However, it will display the participant's image/avatar instead if it is available, such as via [policy profiles](#) or [user records](#). When an audio-only or video-muted Infinity Connect participant is speaking, their indicator expands to show their name in addition to the image or icon.

The name that is shown is decided as described in [Showing the names of active speakers and participants](#) below.







Individual audio-only and video-muted indicators, minimized and expanded

- If there are five or more audio-only or video-muted Infinity Connect participants, they are represented by a single summarized indicator which shows the mix of those participants. When one of the audio-only or video-muted participants speaks, the indicator expands to show their name.



An example of the minimized and expanded audio-only and video-muted indicators when there are 3 audio-only and 2 video-muted participants

- Participants who are receiving video but not sending video (for example, if they have joined a video call without a camera) are represented by an icon of a camera with a line through it .
- Video participants who are on hold, or who are experiencing connectivity issues, are represented by a frozen image.
- If there are more video participants than there are live thumbnail views available, the last thumbnail is replaced by an icon indicating the number of additional participants not currently visible.
- If the conference is being [streamed or recorded](#), a streaming  or recording  icon is displayed. In adaptive layouts this is shown at the top center of the layout. In other layouts it is shown to the right of the main video layout, and whenever a new participant joins the conference, and every two minutes otherwise, the icon will briefly slide out and show its associated text — "Streaming enabled" or "Recording". The icons and text can be changed, and the indicators can be disabled; see [Creating and applying themes to conferences](#).
- When the conference is locked, or has just been unlocked, a lock icon  is displayed to the right of the video window. (This image can be changed; see [Creating and applying themes to conferences](#).)
- If there are participants waiting to join a locked conference, the number of participants is shown beneath the lock icon.

Virtual Meeting Room layouts

All video-enabled participants in a Virtual Meeting Room or scheduled conference see the same layout, regardless of whether they are Hosts or Guests. Administrators can select the layout that will be used for each Virtual Meeting Room, however Hosts can change the layout during the course of a meeting.

To select the layout to be used by default, go to **Services > Virtual Meeting Rooms** or **Services > Scheduled Conferences**, select the Virtual Meeting Room, and then select one of the options from the **View** drop-down menu.

 Host participants using Infinity Connect clients can change the layout during the meeting using the `/layout` [command](#).

The table below lists the layout options that are available. Note that in all cases (except for Adaptive Composition), if one or more participants have been [spotlighted](#), they will take priority over the current speaker.

Option	Command	Description
Adaptive Composition	<code>/layout AdaptiveComposition</code>	Real-time automatic face detection and framing. See Adaptive Composition layout for more information.
Full-screen main speaker only (1 + 0 layout)	<code>/layout 1:0</code>	Participants see a single speaker, with no thumbnails. <ul style="list-style-type: none"> The current speaker is shown full-screen. (The current speaker sees the previous speaker.) The number of other participants (Hosts plus Guests) is indicated in a small icon at the bottom right of the screen.

Option	Command	Description
4 main speakers only (4 + 0 layout)	/layout 4:0	<p>Participants see up to four speakers, with no thumbnails.</p> <ul style="list-style-type: none"> The current speaker plus the three most recent speakers are shown in a 2 x 2 format. (The current speaker sees the four most recent speakers.) The number of other participants (Hosts plus Guests) is indicated in a small icon at the bottom right of the screen. If there are fewer than four other participants: <ul style="list-style-type: none"> a single participant is shown full screen two participants are shown side-by-side three participants are shown in two rows, with one participant in the middle of the top row and the two other participants side-by-side in the bottom row.
Large main speaker and up to 7 other participants (1 + 7 layout)	/layout 1:7	<p>Participants see each other in the standard 1+7 layout.</p> <ul style="list-style-type: none"> The main video shows the current speaker (or previous speaker, for those currently speaking). Up to 7 other participants are shown in a single row of live thumbnails at the bottom of the screen. <ul style="list-style-type: none"> Participants are shown in the thumbnails in order of who spoke most recently, from left to right, regardless of whether they are a Host or Guest. If there are more participants than there are thumbnails available, the number of additional participants is indicated in the far right thumbnail.
Small main speaker and up to 21 other participants (1 + 21 layout)	/layout 1:21	<p>This is the same as the 1 + 7 layout option, except the main speaker is shown slightly smaller in order to accommodate up to 21 other participants in 3 rows of live thumbnails at the bottom of the screen.</p>
2 small main speakers and up to 21 other participants (2 + 21 layout)	/layout 2:21	<p>This is the same as the 1 + 21 layout option, except the top row contains both the main speaker and most recent speaker side by side.</p>
1 small main speaker and up to 33 other participants (1 + 33 layout)	not available	<p>This is a tech preview layout. It has 1 small main speaker at the top center of the layout, with up to 33 other thumbnail participants arranged around it.</p>

Virtual Auditorium layouts

In Virtual Auditoriums, Guest participants usually only ever see the Host participants (with [some exceptions](#)); however, Hosts will see Hosts and Guests. Administrators choose from separate options for the [layout shown to Guests](#) and the [layout shown to Hosts](#) (unless using Adaptive Composition). Additionally, administrators can configure the Virtual Auditorium so that when a presentation is being shown, the [presenter is kept in the main view](#). Host participants can change the layout seen by other Hosts during the course of a meeting.

To select the layouts to be used by default, go to **Services > Virtual Auditoriums**, select the Virtual Auditorium, and then select one of the options from the **Host view** and **Guest view** drop-down menus.

 Host participants using Infinity Connect clients can change the layout during the meeting using the **/layout** [command](#).

Host view

The table below lists the layout options that are available. Note that in all cases, if one or more participants have been [spotlighted](#), they will take priority over the current speaker.

Option	Command	Description
Adaptive Composition	/layout AdaptiveComposition	Real-time automatic face detection and framing. See Adaptive Composition layout for more information. If selected, both Host view and Guest view must use Adaptive Composition.
Full-screen main speaker only (1 + 0 layout)	/layout 1:0	Hosts see a single speaker, with no thumbnails. <ul style="list-style-type: none"> The current Host speaker is shown full-screen. (The currently speaking Host sees the previous Host speaker, or if there are no other Hosts, they see the most recent Guest speaker.) The number of other participants (Hosts plus Guests) is indicated in a small icon at the bottom right of the screen.
4 main speakers only (4 + 0 layout)	/layout 4:0	Hosts see up to four speakers, with no thumbnails. <ul style="list-style-type: none"> The current speaker plus the three most recent speakers are shown in a 2 x 2 format. (The current speaker sees the four most recent speakers.) The number of other participants (Hosts plus Guests) is indicated in a small icon at the bottom right of the screen. If there are fewer than four other participants: <ul style="list-style-type: none"> a single participant is shown full screen two participants are shown side-by-side three participants are shown in two rows, with one participant in the middle of the top row and the two other participants side-by-side in the bottom row.
Large main speaker and up to 7 other participants (1 + 7 layout)	/layout 1:7	Hosts see other Host and Guest participants in the standard 1+7 layout. <ul style="list-style-type: none"> If there is only one Host, they see the most recent Guest speaker in the main video. If there is more than one Host, they see the currently speaking Host in the main video. (The currently speaking Host sees the previous Host speaker.) Up to 7 other participants are shown in a single row of live thumbnails at the bottom of the screen. <ul style="list-style-type: none"> Host participants are shown first in the thumbnails, in order of who spoke most recently, from left to right. Any remaining thumbnails are used to show Guests (again, in order of who spoke most recently). If there are more participants than there are thumbnails available, the number of additional participants is indicated in the far right thumbnail.
Small main speaker and up to 21 other participants (1 + 21 layout)	/layout 1:21	This is the same as the 1 + 7 layout option, except the main speaker is shown slightly smaller in order to accommodate up to 21 other participants in 3 rows of live thumbnails at the bottom of the screen.
2 small main speakers and up to 21 other participants (2 + 21 layout)	/layout 2:21	This is the same as the 1 + 21 layout option, except the top row contains both the main speaker and most recent speaker side by side. This option is useful in combination with the Lock presenter as main speaker option, so that other participants can see the presenter and the person they are speaking to at the same time.
1 small main speaker and up to 33 other participants (1 + 33 layout)	not available	This is a tech preview layout. It has 1 small main speaker at the top center of the layout, with up to 33 other thumbnail participants arranged around it.

Guest view

Guests in a Virtual Auditorium are able to:

- hear and see the Host participant(s)
- hear but not see any of the other Guests, even if a Guest is speaking.

However, there are some circumstances where Guests will be able to see other Guests:

- If all Hosts have left the conference, Guests will be able to see and hear other Guests in the selected layout until the conference is automatically disconnected (after about a minute), or until a Host rejoins.
- If all Hosts are either control-only or audio-only (i.e. they are not sending video), Guests will be able to see and hear other Guests. If a Host then escalates to use video, Guests will then only be able to see the Host.

The table below lists the layout options that are available. Note that in all cases, if one or more participants have been [spotlighted](#), they will take priority over the current speaker.

Option	Description
Full-screen Host speaker only (1 + 0 layout)	<p>Guests see a single Host speaker, with no thumbnails.</p> <ul style="list-style-type: none"> • The Host who spoke most recently is shown full-screen. • If there are two or more Hosts, the number of other Host participants is indicated in a small icon at the bottom right of the screen.
4 main speakers only (4 + 0 layout)	<p>Guests see up to four Host speakers, with no thumbnails.</p> <ul style="list-style-type: none"> • The current Host speaker plus up to the three most recent Host speakers are shown in a 2 x 2 format. • If there are fewer than four Hosts: <ul style="list-style-type: none"> ◦ a single Host is shown full screen ◦ two Hosts are shown side-by-side ◦ three Hosts are shown in two rows, with one participant in the middle of the top row and the two other participants side-by-side in the bottom row. • If there are five or more Hosts, the number of other Host participants is indicated in a small icon at the bottom right of the screen.
Large Host speaker and up to 7 other Hosts (1 + 7 layout)	<p>Guests see Host participants in the standard 1+7 layout.</p> <ul style="list-style-type: none"> • If there is only one Host participant, the Host is shown full-screen. • If there is more than one Host participant, the standard 1+7 layout is used, with the main video showing the currently speaking Host and all other Host participants shown in the thumbnails. • Hosts are shown in the thumbnails in order of who spoke most recently, from left to right. • If there are more Host participants than there are thumbnails available, the number of additional Hosts is indicated in the bottom right thumbnail.
Small Host speaker and up to 21 other Hosts (1 + 21 layout)	<p>This is the same as the 1 + 7 layout option, except the main Host speaker is shown slightly smaller in order to accommodate up to 21 other Hosts in 3 rows of live thumbnails at the bottom of the screen.</p>
2 small Host speakers and up to 21 other Hosts (2 + 21 layout)	<p>This is the same as the 1 + 21 layout option, except the top row contains both the main Host speaker and most recent Host speaker side by side. This option is useful in combination with the Lock presenter as main speaker option, so that other participants can see the presenter and the person they are speaking to at the same time.</p>
1 small main speaker and up to 33 other participants (1 + 33 layout)	<p>This is a tech preview layout. It has 1 small main speaker at the top center of the layout, with up to 33 other thumbnail participants arranged around it.</p>

Lock presenter as main speaker

You can configure a Virtual Auditorium so that when a presentation is being shown, the main speaker position always shows the presenter instead of the current speaker.

Option	Description
Yes	When a presentation is being shown, the selected layout rules are overridden so that: <ul style="list-style-type: none"> The main speaker position always shows the video image from the endpoint that is showing the presentation, even if others are speaking. The image that would have been shown in the main view is instead shown in the first available thumbnail (for 1 + 7 and 1 + 21 layouts) or in the top right view (for 2 + 21 layout).
No	When a presentation is being shown, the main speaker position is voice-switched as usual.

Layout examples

Adaptive Composition layout



Adaptive Composition layout: maximum of 12 video participants are shown, spread across one row of 2 large images, a middle row of 3 slightly smaller images and a bottom row of 7 thumbnail images.

1 + 7 layout



standard 1+7 layout: main video plus a single row of thumbnails, an indicator showing the number of additional video participants, and two icons representing the audio-only participants (one with a personal avatar).

1 + 0 layout



1 + 0 layout: full-screen main video plus an icon indicating the number of additional participants

4 + 0 layout



4 + 0 layout: 2 x 2 main video plus an icon indicating the number of additional participants

1+21 layout



1+21 layout: main video plus three rows of thumbnails, an indicator showing the number of additional video participants, and an indicator showing the total number of audio-only participants.

2 + 21 layout



2 + 21 layout: two main speakers plus three rows of thumbnails, and an indicator showing the number of additional video participants.

1 + 33 layout



1 + 33 layout: one small main video plus up to 33 thumbnails.

Changing aspect ratios

Endpoints send and display video images and presentations in various aspect ratios. The most common ratios are 16:9 and 4:3 (i.e. landscape mode), but mobile devices in particular may send video in portrait mode.

If there is a difference between the aspect ratios of the sending and receiving endpoints, then the receiving endpoint and/or Pexip Infinity may crop the image or add black vertical or horizontal borders (also known as "letterboxing" and "pillarboxing"). Pexip Infinity chooses aspect ratios that maximize interoperability and quality. How and when aspect ratios are changed depends on a number of factors, but the general principles are described below.

Main video

- **Pexip Infinity:** If required, Pexip Infinity will change the aspect ratio of main video by cropping the image (to avoid black bars in the thumbnail images). The exception is when video is received in portrait mode; in this case, the image is not cropped and is displayed as is.
- **Endpoints:** If the aspect ratio of the main video image being sent from Pexip Infinity does not match the aspect ratio of the receiving endpoint, the endpoint will typically add a border to make it fit. Generally endpoints will not crop the image.

Presentations

- **Pexip Infinity:** In general, Pexip Infinity chooses the resolution and aspect ratio that best matches the presentation content and that will not result in both vertical and horizontal borders (also known as "windowboxing") on a 16:9 endpoint display. It will never crop presentations, but may add either pillar or letterboxing to send a resolution that maximizes interoperability.
- **Endpoints:** If the aspect ratio of the presentation image being sent from Pexip Infinity does not match the aspect ratio of the receiving endpoint, the endpoint will typically add another border to make it fit. Generally endpoints will not crop the image.

Limiting the number of participants

You can place a limit on the number of participants that can access a particular Virtual Meeting Room, Virtual Auditorium or scheduled conference at any one time. When such a restriction is in place, when the limit is reached any new participants are presented with a **capacity exceeded** message.

If you change this setting while a conference is in place, participants who have already joined the conference will not be affected, even if the conference is already over capacity. However, after the change has been replicated to all Conferencing Nodes (typically after approximately one minute), any new participants attempting to join the conference will be rejected if the number of participants has reached or exceeded the new limit.

Infinity Connect users who have joined as presentation and control-only participants count towards the participant limits.

Presentation streams from standards-based endpoints and Skype for Business / Lync clients are not counted separately; such participants are only counted once regardless of whether or not there is a presentation stream in addition to the main video.


If you do not restrict the number of participants, any number of participants can join the service (subject to the available [capacity](#) of your Pexip Infinity deployment).

To limit the number of participants in a particular conference:

1. Go to **Services > Virtual Meeting Rooms**, **Services > Virtual Auditoriums** or **Services > Scheduled Conferences** and select the service you wish to change.
2. In the **Advanced options** section, select **Show**.
3. In the **Participant limit** field, enter the maximum number of participants you wish to be able to use the service at any one time.
4. Select **Save**.

Automatically dialing out to a participant from a conference

You can configure a Virtual Meeting Room, Virtual Auditorium, or scheduled conference to automatically dial out to one or more participants whenever a conference using that service starts.

 You can also **manually** dial out to participants from a Virtual Meeting Room, Virtual Auditorium, or scheduled conference on an ad hoc basis. For more information, see [Manually dialing out to a participant from a conference](#).

When is the participant called?

When an Automatically Dialed Participant (ADP) has been added to a Virtual Meeting Room, Virtual Auditorium or scheduled conference's configuration, a call is placed from the conference to the ADP's endpoint as follows:

Meeting type	Participant is automatically dialed...
No PIN	when the first participant joins the conference

Meeting type	Participant is automatically dialed...
Guests not allowed; participants must enter a PIN	when the first participant has entered a valid PIN
Guests allowed; Hosts must enter a PIN but Guests do not	when the first participant joins the conference
Guests allowed; Hosts and Guests must enter a PIN	when the first participant has entered a valid PIN


Note that:

- The role of the first participant does not matter — both Hosts and Guests can trigger a call to an ADP.
- It takes at least 10 seconds (after the first participant joins) for the call to be placed to the ADP.
- A call is not placed to an ADP if that participant has already joined the conference.
- Only one attempt is made to call the ADP — it does not retry.

Host and Guest PINs

When the call is answered, the Automatically Dialed Participant will join the conference as either a Host or Guest, depending on which role you selected when configuring them as an ADP. They will not be required to enter a PIN even if one is required for that role.

Choosing the calling location

 If you have any Pexip Smart Scale locations in your deployment, you should not place calls to ADPs from these locations. See [Configuring Automatically Dialed Participants for PSS](#) for more information.

Usually, the call to the ADP is placed from the same Conferencing Node that is being used by the participant who initiated the conference.

However, you can optionally configure the call to be placed from a Conferencing Node in a specific location. The behavior varies according to whether it is a manually routed ADP (where *Route this call* is set to *Manually*) or the ADP is called according to Call Routing Rules (where *Route this call* is set to *Automatically*).

- **Manually routed ADPs:** the nominated location is the location of the node that will dial the ADP.
- **Automatically routed ADPs:** the nominated location is the notional source location used when considering if a Call Routing Rule applies or not — but the rules themselves will determine the location of the node that dials the ADP. In this case, the configured **Outgoing location** for the ADP is deemed to be the location handling the call (when matched against the **Calls being handled in location** setting for the Call Routing Rule) and the rule's **Outgoing location** setting is used to determine the actual location of the node that dials the ADP.

This behavior is useful if, for example, you have configured lowest-cost-routing rules that only allow calls to be placed from "internal" Conferencing Nodes but you have a conference that is triggered by a call to an "external" Conferencing Node. In this case you still want the call to the ADP to be initiated, but you don't want, in general, to allow calls to arbitrary ISDN destinations to be placed from Conferencing Nodes in the external location.

Incoming call alias

When the endpoint is called, the Automatically Dialed Participant will see the call as coming from one of the Virtual Meeting Room or Virtual Auditorium's aliases. This means that if the participant misses the call, they can easily return it by dialing the alias that appears on their endpoint.

When deciding which alias to use to identify the call, Pexip Infinity will select the first in the list that is valid for the selected call protocol. So if your Virtual Meeting Room has two aliases, the first `meet.sales` and the second `meet.sales@example.com`, when you place a call to an ADP over SIP it will show as coming from `meet.sales` because that is the first valid SIP address in the list. However, endpoints may not always be able to return a call to this alias because it is missing the domain. Therefore, if you want ADPs to be able to return calls to Virtual Meeting Rooms and Virtual Auditoriums, we recommend that you list the most routable aliases first when assigning them to these services. Note that you can also use external or local policy (see [policy profiles](#)) to specify the source alias for calls to ADPs.

Keeping a conference alive

Automatically Dialed Participants with a role of Host may or may not prevent a conference from being automatically terminated, depending on their **Keep conference alive** setting and whether any other ADPs remain in the conference. For more information, see [Automatically ending a conference](#).

Bandwidth limits

When dialing out from a conference, the outbound call bandwidth limit is inherited from the VMR's bandwidth settings. (If a Call Routing Rule is applied, the rule's bandwidth settings are not used.)

Configuration

i If you have a large number of ADPs to configure, you can import them using a CSV file. For more information, see [Bulk import/export of service configuration data](#).

To configure a Virtual Meeting Room or Virtual Auditorium to automatically dial out to a participant when a conference starts:

1. Go to **Services > Automatically Dialed Participants**.
2. Select **Add Automatically Dialed Participant**.
You will be taken to the **Add Automatically Dialed Participant** page.
3. Complete the following fields:

Field	Description
Participant alias	The alias of the participant to dial when a conference starts. If you select a Protocol of <i>SIP</i> , this must be a valid SIP alias.
Creation time	This read-only field shows the date and time when this record was first configured.
Participant display name	An optional user-facing display name for this participant, which may be used in participant lists and as the overlaid participant name (if enabled). If this name is not specified then the Participant alias is used as the display name instead.
Description	An optional description of the Automatically Dialed Participant.
Route this call	Select how to route the call: <ul style="list-style-type: none">◦ Manually: uses the requested Protocol and the defaults for the specified System location.◦ Automatically: routes the call according to the configured Call Routing Rules. This means that the dialed alias must match an outgoing Call Routing Rule for the call to be placed (using the protocols, outgoing location and call control systems etc. as configured for that rule). Default: Manually .
Protocol	The signaling protocol to use when dialing the participant. Select either <i>SIP</i> , <i>H.323</i> , or if the endpoint is a Skype for Business / Lync client, select <i>Lync / Skype for Business (MS-SIP)</i> . The RTMP protocol is typically used when adding a streaming participant. Note that if the call is to a registered device, Pexip Infinity will instead use the protocol that the device used to make the registration. Default: <i>SIP</i> . This field only applies when Route this call is set to Manually .

Field	Description
DTMF sequence	<p>An optional DTMF sequence to be transmitted after the call to the Automatically Dialed Participant starts.</p> <p>A DTMF sequence can include: the digits 0-9, "*" (asterisk), "#" (hash) or "," (comma).</p> <p>The DTMF tones are sent 3 seconds after the call connects, one at a time, every 0.5 seconds. A comma is a special digit that represents a 2 second pause (multiple commas can be used if a longer pause is needed).</p> <p>For example, if you need your Automatically Dialed Participant to dial an audio bridge and then enter conference number 777 followed by #, pause for six seconds and then supply conference PIN 1234 followed by #, you would configure 777#,,,1234# as your DTMF sequence.</p>
Role	<p>The level of privileges the participant will have in the conference. For more information, see About PINs, Hosts and Guests.</p> <p>Default: <i>Guest</i>.</p>
Conference	<p>Select the names of the Virtual Meeting Rooms, Virtual Auditoriums and scheduled conferences from which this participant will be dialed automatically whenever a conference using that service starts.</p>
Outgoing location	<p>For Manually routed ADPs, this is the location of the Conferencing Node from which the call to the ADP will be initiated.</p> <p>For Automatically routed ADPs, this is the notional source location used when considering if a routing rule applies or not - however the routing rule itself determines the location of the node that dials the ADP. For more information, see Choosing the calling location.</p> <p>To allow Pexip Infinity to automatically select the Conferencing Node to use to place the outgoing call, select Automatic.</p> <p>As with all calls, signaling and media may be handled by different Conferencing Nodes in that location.</p>
Streaming	<p>Identifies the dialed participant as a streaming or recording device. When a conference participant is flagged as a streaming/recording participant, it is treated as a receive-only participant and is not included in the video stage layout seen by other participants. See Streaming and recording a conference for more information.</p>
Keep conference alive	<p>Determines whether the conference will continue when all other non-ADP participants have disconnected:</p> <ul style="list-style-type: none"> Yes: the conference will continue to run until this participant has disconnected (applies to Hosts only). If multiple: the conference will continue to run as long as there are two or more If multiple participants and at least one of them is a Host. No: the conference will be terminated automatically if this is the only remaining participant. <p>Default: <i>If multiple</i>.</p> <p>For streaming participants, we recommend that this option is set to <i>No</i>.</p> <p>For more information, see Automatically ending a conference.</p>
Call capability	<p>Allows you to limit the media content of the call. The participant being called will not be able to escalate beyond the selected capability. For more information, see Controlling media capability.</p> <p>Default: Main video + presentation.</p> <p>This field only applies when Route this call is set to Manually.</p> <p>This field does not apply to RTMP calls.</p>
Dual stream (presentation) URL	<p>When adding a dual streaming RTMP participant, this specifies the RTMP URL for the second (presentation) stream.</p> <p>Leave this field blank when adding a single streaming participant.</p>

4. Select **Save**.

Automatically ending a conference

You may want to ensure that conferences are automatically terminated in certain situations, in order to preserve resources and to ensure that certain types of participants can't keep the conference alive by remaining in a call indefinitely. For this reason, there are settings that allow you to end a conference in any of the following situations:

- when [only Guests remain](#)
- when there is [only one participant remaining in the conference](#)
- when the [only participants remaining are ADPs or participants that have been added by an administrator](#).

These settings are all independent; a conference needs to meet only one of the above criteria in order for it to be terminated automatically on that basis. For example, you may have configured your system to terminate conferences 120 seconds after the last Host leaves, and 60 seconds after there is only one participant remaining. In that case:

- if all but one Guest remains in a conference 30 seconds after the last Host has left, the conference will be terminated 60 seconds later (i.e. based on the one participant remaining setting)
- if all but two Guests remain 30 seconds after the last Host has left, the conference will be terminated in another 90 seconds (i.e. based on the last Host setting).

When only Guests remain

Pexip Infinity will always terminate a conference a certain period after the last Host disconnects and only Guests remain in the call. By default this period is set to one minute, but it can be set to between 0 seconds (i.e. all Guests are disconnected immediately upon the last Host disconnecting), and one day.

The default is 60 seconds. This should be sufficient time for a Host who has been unintentionally disconnected from a conference to reconnect before the conference is terminated.

To change this setting, go to **Platform > Global Settings > Service Configuration > Guests-only Timeout**.

When there is only one participant remaining in the conference

You can configure the length of time (in seconds) for which a conference will continue with only one participant remaining. This can be set to between 60 seconds and one day, or alternatively you can set it to 0 which means that a conference will never be terminated on the basis that there is a single participant remaining. The default is 0.

When this setting is used to terminate a conference, the type of participant (Host, Guest, ADP, administrator-added, streaming, etc.) is irrelevant; if they are the only participant remaining, the conference will be terminated after the specified time.

To change this setting, go to **Platform > Global Settings > Service Configuration > Last Participant Backstop Timeout**.

When the only participants remaining are ADPs and/or administrator-added

In most cases a conference won't be terminated if a Host participant is present. However, if the only remaining Host participants are:

- Automatically Dialed Participants (ADPs), and/or
- participants who have been added to a conference by an administrator (either from the Administrator interface or using the Management API),

then the conference may or may not be terminated, depending on the participants' **Keep conference alive** settings and whether any other ADPs or administrator-added participants remain in the conference.

Keeping a conference alive

There are three options for the **Keep conference alive** setting:

- **Yes:** if the participant is a Host, the conference will continue to run even when this is the only participant remaining - in other words, they are just like any other Host participant. This is the default used when adding a participant to a conference [using the Administrator interface](#) or using the Management API.
- **No:** the conference will be automatically terminated if this is the only participant remaining. This is the option we recommend for automated systems that are unable to terminate a call themselves, such as streaming participants.
- **If multiple:** if two or more ADP or administrator-added participants with the **If multiple** setting remain in the call and at least one of them is a Host, the conference will not be terminated by Pexip Infinity. However, if all other participants have disconnected and

only one ADP or administrator-added participant with the *If multiple* setting remains, the conference will be terminated by Pexip Infinity. This is to prevent automated systems (such as recording devices) that are unable to terminate a call themselves from keeping the conference alive indefinitely. For this reason, if you are using this option we recommend that each Virtual Meeting Room or Virtual Auditorium has no more than one such automated system as an ADP. A better alternative is to give automated systems a **Keep conference alive** setting of *No* (see above).

If multiple is the default option when [adding an ADP](#).

If the only remaining participants in a conference are ADPs and/or administrator-dialed participants with a mix of *Yes*, *If multiple* and *No* options, the conference will be terminated unless:

- at least one participant has a **Keep conference alive** setting of *Yes*, or
- there are two or more participants with a setting of *If multiple* and at least one of them is a Host.

Controlling media capability

Calls consist of three types of media - audio, video, and presentation - each in their own stream. Pexip Infinity allows you to control which types of media can be used for certain calls. This feature allows better resource management and smaller Session Description Protocols (SDPs) when calling out to known audio-only devices.

Limiting media capability

You can restrict the media capability either on a per-call basis (the call capability) or a per-conference basis (the conference capability), as follows:

Adding a participant to a conference

When dialing out to a participant from within a conference, either [manually](#) or [automatically](#), you can choose whether the call will be **audio-only**, **main video only**, or **main video plus presentation**. In such cases, the called participant will not be able to escalate the call (for example, from an audio-only call to video).

Conference-wide limitations

You can apply limits to individual [Virtual Meeting Rooms](#) (including scheduled conferences) or [Virtual Auditoriums](#), restricting them to be **audio-only**, **main video only**, or **main video plus presentation**. In such cases, participants calling in to the conference with a higher media capability will have their media limited. For example, when a participant dials in over video to an audio-only VMR, their call will be placed as audio-only and they will not subsequently be able to escalate the call.

Using the **audio-only** setting where appropriate also has the advantage of increasing the overall capacity of your distributed deployment. Each Conferencing Node will reserve an extra connection for each Virtual Meeting Rooms and Virtual Auditoriums it is hosting, to be used for a backplane should that service become distributed (i.e. hosted on more than one Conferencing Node). By default this connection will be equivalent in capacity to an HD call, but if the service has been configured as **audio-only** it will instead be equivalent to that of an audio-only call.

Virtual Receptions

You can limit the conference capability of a [Virtual Reception](#). Any restrictions will be applied to calls while they are accessing the Virtual Reception service (apart from Infinity Connect clients, which do not access a Virtual Reception in the same way as other endpoints). When the call is placed to the destination Virtual Meeting Room or Virtual Auditorium, the capability of that service will be available instead, but may or may not be used by the endpoint, as follows:

- Infinity Connect clients making a video call via an audio-only Virtual Reception will always join a video-capable VMR with video
- Infinity Connect clients making an audio-only call via an audio-only Virtual Reception will join a video-capable VMR with audio-only initially, but with the option of enabling video.
- Lync / Skype for Business clients making a call via an audio-only Virtual Reception will always join a video-capable VMR as audio-only initially, but with the option of enabling video.
- Standards-based endpoints making a video call via an audio-only Virtual Reception will always join a video-capable VMR as audio-only initially. Depending on the endpoint, there may be the option to then enable video.

For example, if you limit your Virtual Reception to audio-only, when Alice calls it from her standards-based endpoint she will not receive any video and will only hear the audio prompts. However, when her call is then placed to the destination VMR which has a capability of **main video**, she will initially join it as audio-only but will be able to elect to send and receive video as well.

Infinity Gateway calls

For calls made using the [Distributed Gateway](#), you can configure the relevant Call Routing Rule to limit the media by selecting an appropriate **Call capability** setting:

- You can limit the media capability of the outbound call to **audio-only**, **main video only**, or **main video plus presentation**. For example, if the media capability is set to **audio-only** and Alice makes a video call to Bob, Bob will not be able to answer with video.
- You can match the capability of the outbound call to that of the inbound call by using the **Same as incoming call** option. This means that if, for example, Alice makes an audio-only call to Bob, Bob will not be able to answer with video.

Note that, when using **Same as incoming call**:

- an audio-only call cannot later be escalated to video
- auto-escalation of Skype for Business / Lync calls (if enabled) will not work
- for an H.323 audio-only call to a Skype for Business / Lync client, the Skype for Business / Lync client will be offered both audio and video.

Automatically escalating Skype for Business / Lync audio calls




You can control how Pexip Infinity handles incoming audio-only calls from Skype for Business / Lync* clients.

By default, Pexip Infinity treats the call as audio-only and does not send video back to the SfB/Lync participant.

This behavior can be changed on a platform-wide basis by selecting the **Enable Skype for Business / Lync auto-escalation** setting from **Platform > Global Settings > Connectivity**.

When SfB/Lync auto-escalation is enabled, Pexip Infinity automatically escalates a SfB/Lync audio-only call so that it receives video from a conference (the SfB/Lync participant still only sends audio).


The following table shows the differences in conference experience based on the auto-escalation setting when a SfB/Lync client makes an audio call to a Pexip conference:

SfB/Lync auto-escalation configuration		
	Disabled (default)	Enabled
The SfB/Lync participant...	sees the conference avatar image ( by default)	receives a video stream from the conference
Other conference participants see the SfB/Lync participant as...	an audio-only participant indicator on the left side of the video window ( by default)	a "no incoming video" indicator when the participant is shown as the main speaker or as a thumbnail ( by default)

In all cases, the SfB/Lync participant can still subsequently escalate to send (and receive) video.

* Note that where this documentation refers to "SfB/Lync", it represents both Microsoft Skype for Business and Lync unless stated otherwise.

Streaming and recording a conference

 This topic provides a general overview of Pexip Infinity's support for streaming and recording. For specific step-by-step instructions on how to integrate with some popular streaming services, see [Integrating with streaming and recording services](#).

Pexip VMRs can output a dedicated RTMP/RTMPS multimedia stream to enterprise CDN (Content Delivery Network) streaming and recording services such as Wowza, Quickchannel, Qumu, VideoTool, Microsoft Stream and Azure Media Services, and to public streaming services such as YouTube, Facebook and Periscope. Any Pexip conference can be streamed as a live event to an unlimited number of viewers, and can automatically be recorded and stored for later consumption.

If your conference contains presentation content, you have the option of setting up dual streams so that you can output the main video and presentation content channels separately. This means that viewers can simultaneously access the main video stream of the participants and a separate presentation content stream.

The administrator or meeting Host has to decide whether to set up the conference with a single RTMP stream or with dual RTMP streams. Currently, you cannot set up a single stream purely for presentation content.

When a conference is single streamed:

- Viewers are automatically switched between the main video stream and the presentation stream whenever someone is presenting.
- When someone is presenting, the video stream of the active speaker is shown in a small window in the upper right corner of the presentation. Currently, you cannot set up a single stream purely for presentation content (i.e. without the small video window).
- The same single stream (with automatic switching between main video and presentation streams) can be made available later as a recording.



When a conference is dual streamed:




- Viewers have to manually switch between the main video stream and the presentation stream, according to their preference. (However, viewers could optionally view the streams in two separate windows — one window showing the main video stream and the other window showing the presentation stream.)
- The main video stream is not embedded as a small window within the presentation stream.
- If nobody is currently presenting, Pexip Infinity sends a placeholder image on the presentation stream.
- If the event is made available later as a recording, it can only be played back as two completely separate streams. There is no synchronization between the streams.


Note that your firewall needs to allow outbound traffic from the Conferencing Node to TCP port 1935 on the RTMP streaming server.

If required, you can set up multiple streams from the same conference to different streaming services.

Streaming/recording indicators and announcements

If the conference is being streamed or recorded, a streaming  or recording  icon is displayed. In adaptive layouts this is shown at the top center of the layout. In other layouts it is shown to the right of the main video layout, and whenever a new participant joins the conference, and every two minutes otherwise, the icon will briefly slide out and show its associated text — "Streaming enabled" or "Recording". The icons and text can be changed, and the indicators can be disabled; see [Creating and applying themes to conferences](#).

- The streaming indicator  is displayed when a streaming participant is added to the conference.
- The recording indicator  is displayed if a Microsoft Teams or Google Meet conference is being recorded, or a Skype for Business / Lync client records a conference.
- The participant list on Infinity Connect clients also shows the streaming or recording device to which the stream is being sent. A streaming participant has a streaming badge  next to its name (which usually takes the form of the URL to which the stream is being sent).
- Note that an indicator is shown to SfB/Lync clients if streaming is initiated by Pexip Infinity.

For administrators looking at a conference graph, streaming participants are identified by a  indicator.

Announcements to participants in Google Meet and Microsoft Teams conferences

- If a Google Meet conference is recorded or streamed, audio prompts indicating that streaming or recording has been started/stopped are played to callers who are gatewayed via Pexip Infinity into the conference, and distinct messages and indicators are used depending on whether the conference is being recorded, streamed or both. When streaming, the audio prompts and indicators also vary according to whether the stream is public or not.
- If a Microsoft Teams conference is recorded or transcribed, relevant audio prompts indicating that recording/transcribing has been started/stopped are played to callers who are gatewayed via Pexip Infinity into the conference.

The content of these audio files can also be changed by customizing a theme.

Initiating streaming from an Infinity Connect client

To initiate streaming from an Infinity Connect client, you add the streaming or recording device as a participant in the VMR.

When an Infinity Connect client adds a new participant to a conference, **Automatic routing** is used, meaning that the dialed alias must match an appropriate [Call Routing Rule](#), and the call is then placed using the protocols and other settings as specified in the rule.

A suitable rule to match rtmp and rtmps aliases would typically include the following settings:

Option	Setting
Name	A suitable description such as "Streaming to <streaming service>".
Priority	This is a very specific rule, so we recommend that you give it a relatively high priority (a low number).
Incoming gateway calls	If you want to use this rule just for calls placed from a conference, then leave this blank (disabled). However, if you also want to use this rule to enable incoming gateway calls to be placed out to streaming or recording devices, then enable this option.
Outgoing calls from a conference	Enabled
Match against full alias URI	Enabled
Destination alias regex match	rtmps?:/(.*)
Destination alias regex replace string	Leave this option blank — we want to dial the alias as it was received.
Outgoing location	Ensure that you select a location that is able to place calls to the streaming or recording system, such as a location containing Proxying Edge Nodes when streaming to external RTMP ingest endpoints.
Protocol	RTMP (streaming)

After you have created the above rule, it will also apply to any calls placed [using the administrator interface](#), or to an [Automatically dialed participant](#), where you have selected to route the call automatically.

Alternatively, you can bypass the need for call routing rules by [using the administrator interface](#) to dial out to the streaming participant, as follows:

- **Participant alias:** the address of the streaming service (typically prefixed with `rtmp://` or `rtmps://`)
- **Route this call:** *Manually*
- **Protocol:** *RTMP*
- **Streaming:** enabled.

In this case, because the call is routed manually, Call Routing Rules are not required.

Streaming when Pexip Infinity enforces media encryption

Some streaming services such as YouTube support RTMP only, and not RTMPS. Therefore, if [Media encryption](#) is set to *Required* (either globally or for the specific VMR being used), a call to the RTMP-only streaming service will fail.

In such cases, if appropriate, you may consider changing the Media encryption setting for the VMR to *Best effort*, which will allow outbound calls to be placed over RTMP.

General instructions for setting up streaming

In general, to integrate with a third-party streaming or recording service:

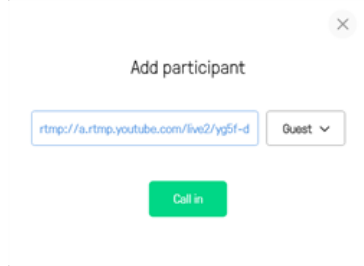
1. From the streaming provider, obtain an address to which the video stream will be sent.
2. Initiate a call from the conference to the streaming address. This is done by adding the streaming address as a conference participant. You can do this either from the [Pexip Infinity Administrator interface](#) or from an [Infinity Connect client](#) connected to the VMR. Alternatively, for services that offer persistent URLs (such as with Periscope or YouTube) which therefore can be re-used for subsequent streams, you could set up the URL to be [automatically dialed](#) whenever a particular VMR is used.



When using the Administrator interface, use the following settings:

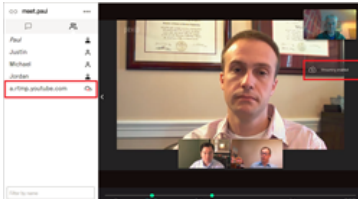
- **Protocol:** *RTMP*
- **Address:** the address obtained from the streaming provider.
- **Role:** we recommend selecting *Guest* (so that the streaming participant is not shown to other Guests in a Virtual Auditorium layout, and so that it does not keep a conference alive when all other Hosts have left).
- **Streaming:** select this option.


When using an Infinity Connect client, use the following settings:

- **Participant details:** enter your rtmp/rtmps alias e.g. `rtmp://a.rtmp.youtube.com/live2/yg5f-dkm5-vm27-0kw6`
RTMP authentication is supported; in this case credentials are included in the URI using the syntax `rtmps://username:password@host/...`
Note that a [suitable Call Routing Rule](#) is required when dialing out to a streaming service via Infinity Connect clients.
- **Role:** we recommend selecting *Guest*.



3. When Pexip Infinity has placed the call to the streaming service, the **Streaming enabled** icon  is displayed, and for Infinity Connect users the streaming participant appears in the participant list with a streaming badge  next to its name:



-  Skype for Business / Lync clients can also use the Infinity Gateway service to dial out to an RTMP streaming or recording service from within a Skype for Business / Lync meeting.

Setting and limiting call quality

Many factors can affect the video call quality as seen by individual participants on a call. Some of these factors are external to Pexip Infinity, such as the call protocols used by the endpoints participating in the conference, the compute and camera resources available to the endpoint, and constraints introduced by the network or call control systems. However, other factors that influence quality can be controlled within Pexip Infinity, typically at the platform level but with the ability to override those global settings for specific conferences.

Video call quality typically depends upon the resolution (for sharpness, and often expressed in terms of SD, HD or Full HD) and the frame rate (for smoothness). High resolutions and frame rates require more bandwidth than lower resolutions and frame rates, although the exact requirements vary according to the codec and compression algorithms being used.

The choice of resolution, frame rate and codec also impacts the amount of compute resource required by a Conferencing Node to host a conference, and hence the overall capacity of each Conferencing Node in terms of the number of concurrent participants and conferences it can host.

It is important to note that endpoints ultimately decide what bandwidth and resolution they send to Pexip Infinity, while Pexip Infinity is responsible for deciding what gets sent to the endpoints.

Maximum call quality

While the actual quality of the call as seen by individual participants depends upon the factors discussed above, you can configure within Pexip Infinity a **Maximum call quality** that limits what a Conferencing Node will send to — and request from — each participant in the conference.

The **Maximum call quality** options are SD, HD or Full HD and the associated resulting maximum resolution and frame rate for video are shown below:

Maximum call quality (video resolution)	Maximum resolution	Maximum frame rate (fps)
SD (448p)	768 x 448	30
HD (720p)	1280 x 720	30
Full HD (1080p)	1920 x 1200	30

By default, Pexip Infinity conferences have a maximum call quality of HD. You can configure this at the global platform level and, if required, override it for each individual service (VMR, Call Routing Rule and so on). For example, you could use the default option of "HD" for most of your services by default, but enable Full HD on some specific services.

Note that this is the **maximum** quality that Pexip Infinity will send to conference participants. The configured **Maximum outbound call bandwidth** for a service can cause Pexip Infinity to select a lower quality than the configured **Maximum call quality** (see [Managing and restricting call bandwidth](#) for more information).

Impact on resource usage

The **Maximum call quality** setting also controls how much compute resource is allocated and reserved by a Conferencing Node for each participant that joins the conference. This is measured within Pexip Infinity in relation to the amount of resources required by a standard HD connection.

In general, when compared to a single high definition **HD 720p** call:

- a **Full HD 1080p** call uses twice the resource
- an **SD** standard definition call uses half the resource
- an **audio-only** call uses one twelfth of the resource.

A WebRTC call using the **VP8** codec uses the same amount of resource as H.264, and the **VP9** codec uses around 25% more resource, so VP9 at 720p uses the equivalent of 1.25 HD resources, and VP9 at 1080p uses the equivalent of 2.5 HD resources. WebRTC clients also use 0.5 HD additional resources for sending presentation content and 1 additional HD resource when receiving full motion presentation. Note that within the same conference some participants may use VP9 (if they are connected to a Conferencing Node using the AVX2 or later instruction set) while other participants may use VP8 (if they are connected to a Conferencing Node on older hardware).

Conferences that use the **Adaptive Composition** layout consume additional Conferencing Node resources. The actual amount of additional resource depends on many factors, but as a guide, it uses an additional 0.5 HD of resource for each video participant that is on stage (up to 13 participants per conference). This is regardless of the call quality / resolution of the conference itself and each individual participant's connection (codec, bandwidth and so on).

Thus, setting the maximum call quality to a "high" value such as Full HD will result in more resources being reserved than selecting a "low" value such as SD, and the more resources that are used or reserved means a lower capacity in terms of overall concurrent connections (also referred to as ports) for each Conferencing Node.

Considerations for using Full HD (1080p) for main video in calls

Enabling Full HD (1080p) capabilities allows any endpoint capable of Full HD to send and receive its main video at 1080p to those conferences. However, as discussed above, enabling Full HD has implications on bandwidth and capacity across your deployment, specifically:

- Full HD calls require approximately double the Conferencing Node resources and double the bandwidth of an HD call.
- 1 Full HD of capacity will be reserved for backplanes between Conferencing Nodes.

Note that 1080p is automatically used for sharing high-resolution content with HD-capable endpoints if there is sufficient available bandwidth i.e. presentation content may still be sent at 1080p even if Full HD is not allowed for main video.

Setting the maximum call quality for participants in a conference

To set the maximum call quality for all calls across your entire deployment:

1. Go to **Platform > Global Settings**.
2. In the **Service Configuration** section, select the required **Maximum call quality**:

- **SD**: each participant is limited to SD quality.
- **HD**: each participant is limited to HD (720p) quality.
- **Full HD (1080p)**: allows any endpoint capable of Full HD to send and receive its main video at 1080p.

3. Select **Save**.

To override the global default and set the maximum call quality for an individual service or Call Routing Rule:

1. Go to the relevant service or rule:
 - **Services > Virtual Meeting Rooms**
 - **Services > Virtual Auditoriums**
 - **Services > Virtual Receptions**
 - **Services > Scheduled Conferences**
 - **Services > Call Routing**
2. Either select the name of the service or rule you want to edit, or click **Add**.
3. In the **Advanced options** section, or the **Call media settings** section for routing rules, select the required **Maximum call quality**:
 - **Use global setting**: use the global maximum call quality setting.
 - **SD**: each participant is limited to SD quality.
 - **HD**: each participant is limited to HD (720p) quality.
 - **Full HD (1080p)**: allows any endpoint capable of Full HD to send and receive its main video at 1080p.
4. Select **Save**.

Changes to the maximum call quality take effect for any new conferences initiated after the change has been made.

Managing and restricting call bandwidth


You can restrict the amount of bandwidth used by each participant when dialed in to a Pexip Infinity service (Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service), or for Infinity Gateway calls or other outbound calls that are managed by Call Routing Rules. You can place restrictions on the bandwidth received by participants, sent by participants, or both.

Bandwidth restrictions can be configured on a [global basis](#), or on each individual [service or routing rule](#). Settings applied to an individual service override any global limits that have been applied.

When sending main video and presentation to a standards-based endpoint, you can also control the maximum percentage of the call bandwidth to allocate to the presentation content (see [presentation bandwidth requirements](#)).

If a Virtual Reception has bandwidth limits, these only apply to participants while they are in the Virtual Reception. When the call is transferred to the selected Virtual Meeting Room or Virtual Auditorium, any bandwidth restrictions for that service will then apply.

When dialing out from a conference, the outbound call bandwidth limit is inherited from the VMR's bandwidth settings. (If a Call Routing Rule is applied, the rule's bandwidth settings are not used.)

 Inbound bandwidth restrictions are implemented as requests to the participant endpoints to limit their bandwidth to the specified amount. It is important to note that endpoints ultimately decide what bandwidth and resolution they send — Pexip Infinity cannot actually enforce inbound restrictions. However, Pexip Infinity is responsible for deciding what gets sent to the endpoints.

Restricting video call resolutions

If you want to limit video calls to specific resolutions (and limit the transcoding node resources that are reserved for calls), you should use the **Maximum call quality** setting.

However, bandwidth settings also affect the call resolution as shown below. If there is insufficient bandwidth (the **Maximum bandwidth** settings configured against the service) to support the **Maximum call quality** then the participant's call will use a lower resolution as appropriate for the available bandwidth (and consume less transcoding node resource).

Minimum call bandwidth requirements

This table shows the minimum bandwidth required for Pexip Infinity to be able to send different video resolutions. Note that this **does not** include the audio component, which can add between 8 kbps and 64 kbps.

Video resolution	Minimum required bandwidth
Full HD (1080p)	2400 kbps (1600 kbps for VP9)
HD (720p)	960 kbps (640 kbps for VP9)
SD (448p)	448 kbps
SD (QCIF)	64 kbps

Note that when compared to standards-based endpoints, the WebRTC VP9 codec provides the same resolution at a lower bandwidth, but consumes around 25% more resource on the Conferencing Node. This is why you should use **Maximum call quality** rather than bandwidth restrictions to limit [resource consumption](#).

In most cases you should apply the same bandwidth restrictions to the inbound and outbound calls within a service. However, for example, if you want to allow participants to send their video as SD, but receive the composed layout of all participants (main video and video thumbnails) as HD, you would set the inbound call bandwidth to 960 kbps and the outbound to 1800 kbps (although these settings would not limit clients using VP9).

Applying bandwidth restrictions to an entire deployment

To restrict the bandwidth of calls across your entire deployment:

1. Go to **Platform > Global Settings**.
2. In the **Service Configuration** section, enter the desired values in the following fields:

Option	Description
Maximum inbound call bandwidth (kbps)	Limits the bitrate of media received by Pexip Infinity from a participant. Leave blank if you do not want to apply any restrictions. This can be overridden by any services or rules that have a specific Maximum inbound call bandwidth configured.
Maximum outbound call bandwidth (kbps)	Limits the bitrate of media sent from Pexip Infinity to a participant. Leave blank if you do not want to apply any restrictions. This can be overridden by any services or rules that have a specific Maximum outbound call bandwidth configured.

Applying bandwidth restrictions to a service or Call Routing Rule

To restrict the bandwidth of calls to a particular service or Call Routing Rule:

1. Go to the relevant service or rule:
 - **Services > Virtual Meeting Rooms**
 - **Services > Virtual Auditoriums**
 - **Services > Virtual Receptions**
 - **Services > Scheduled Conferences**
 - **Services > Call Routing**
2. Either select the name of the service or rule you wish to edit, or click **Add**.
3. In the **Advanced options** section, select **Show** (not required for Call Routing Rules).

4. Enter the desired values in the following fields:

Option	Description
Maximum inbound call bandwidth (kbps)	Limits the bitrate of media received by Pexip Infinity from a participant. Leave blank if you do not want to apply any restrictions. This overrides any global Maximum inbound call bandwidth that may have been configured.
Maximum outbound call bandwidth (kbps)	Limits the bitrate of media sent from Pexip Infinity to a participant. Leave blank if you do not want to apply any restrictions. This overrides any global Maximum outbound call bandwidth that may have been configured.

5. Select **Save**.

Presentation bandwidth requirements

Bandwidth usage for presentation streams depends on the type of client:

- When sending main video and presentation content to a conference participant on a standards-based (SIP or H.323) endpoint, Pexip Infinity splits the available bandwidth between main video and content.
 - By default up to 75% of the available call bandwidth is allocated to presentation content (and thus 25% is allocated to main video). You can modify this ratio at the global platform level via **Platform > Global Settings > Service Configuration > Maximum Presentation Bandwidth Ratio**.
 - The resolution and frame rate used for presentation content adapts dynamically to match the presented content, and the **Maximum call quality** is used to determine the maximum frame rate that Pexip Infinity will send for a given resolution. For example, if the maximum call quality is HD, then the maximum frame rate for Full HD content is 13 fps, but if an incoming presentation is Full HD at 4 fps then Pexip Infinity will send 4 fps.
- When using a WebRTC-based Infinity Connect client to present content or to receive full motion (HD) presentation, the bandwidth is not shared. The presentation stream is treated as a separate call and has the same bandwidth limit applied as the main video call.
- When sending main video and presentation content to a Skype for Business / Lync client, the bandwidth is not shared (this applies to VbSS and RDP). The presentation stream is treated as a separate call and has the same bandwidth limit applied as the main video call.
- When sending presentation content to Microsoft Teams, Pexip always sends content to Teams at Full HD (1080p) at a frame rate requested by Teams. The resolution and frame rate of content sent from Microsoft Teams is determined by Teams.
- When sending main video and presentation content to a Google Meet meeting, the presentation stream is treated as a separate call and has the same bandwidth limit applied as the main video stream. When sending video or presentation content to Pexip Infinity, Google Meet does not have any fixed bandwidth limitations — it dynamically downscales and upscales individual calls in response to network conditions, although it typically will not exceed 2 Mbps.

Enabling and disabling chat messages

Conference participants who use a chat-enabled client can send messages and share links with each other within a Virtual Meeting Room or Virtual Auditorium, and when calling each other directly via the Infinity Gateway. Supported clients include Microsoft Skype for Business and Lync clients and Pexip's own Infinity Connect suite.

Chat support is configurable on a platform-wide and per-conference basis, and is enabled by default.

To configure chat at the platform level:

- Go to **Platform > Global Settings**.
- From within the **Connectivity** section, deselect or select **Enable chat**.

You can also override the global setting on a per conference basis if required. To do this:

- Go to **Services > Virtual Meeting Rooms**, **Services > Virtual Auditoriums** or **Services > Scheduled Conferences**.
- From within the **Advanced Options** section, choose one of the **Enable chat** options:

- **Use global chat setting:** as per the [global configuration setting](#).
- **Yes:** chat is enabled.
- **No:** chat is disabled.

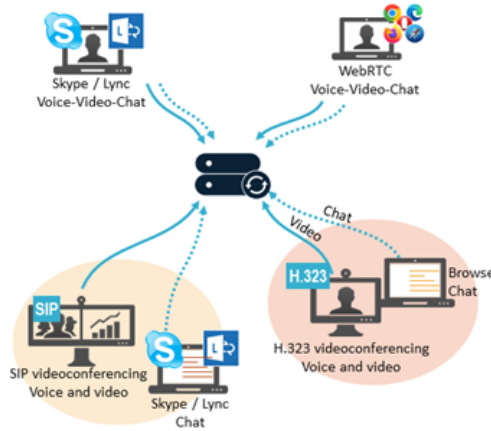
Default: **Use global chat setting.**

When chat is disabled, Infinity Connect clients do not show the chat window.

Providing chat to participants using unsupported clients

Conference participants who are not using one of the supported clients are not able to read or send chat messages.

However, if they have access to a web browser they can use the Infinity Connect web app to join the conference without video or audio. This will give them access to the chat room, plus the ability to view and share presentations, view the participant list, and (if they are Host participants) control aspects of the conference.



For more information on using and administering the Infinity Connect suite of clients, see [Introduction to Infinity Connect](#).

Playing notification tones when participants join or leave a conference

Pexip Infinity can automatically play a sound when a participant joins or leaves a conference. These entry and exit tones are disabled by default, but you can enable them on selected services using [themes](#).

Enabling notification tones

Entry and exit tones are applied using themes. To enable entry or exit tones for a particular Virtual Meeting Room or Virtual Auditorium, you must ensure that the theme used for that service includes the following files:

- **conf-participant_entry_tone_48kHz_mono.wav**, containing the sound to play when a participant joins the conference.
- **conf-participant_exit_tone_48kHz_mono.wav**, containing the sound to play when a participant leaves the conference.

By default, entry and exit tones are not played, as the Base theme contains "empty" entry and exit tone files. However, Pexip Infinity also ships with a selection of [preconfigured](#) themes, some of which contain entry and exit tones in the .wav files listed above.

This means that to enable notification tones you can do any of the following:

- select one of the preconfigured themes containing tones
- use the .wav files from those preconfigured themes in your own themes
- produce your own .wav files and include those files in your own themes.

Disabling notification tones

The Base theme contains empty entry and exit tone files, and is shipped as the default theme. This means that by default, entry and exit tones will not be played. You can [download](#) and copy the empty entry and exit tone files from the Base theme and use them in any custom themes you create for which you don't want to include entry and exit tones. Some of the preconfigured themes also contain empty entry and exit tone files.

If you apply a theme that does not include these files (rather than including empty files), Pexip Infinity will instead use the entry and exit tone files from the theme that has been selected as the [default theme](#).

Therefore, if you want to ensure that entry or exit tones are not played, you must still include the entry tone and exit tone files in the theme, but leave them empty.

Controlling active conferences


Locking a conference and allowing participants to join a locked conference

You can lock a conference if you want to prevent any further participants from joining a conference after it has started. A conference can be locked and unlocked by conference participants [using Infinity Connect](#) or [using DTMF-enabled endpoints](#), or by [using the Administrator interface](#).


When a conference is locked, any new participants who attempt to join the conference are held at a waiting screen. They can be [allowed in individually](#) by Infinity Connect participants (Hosts only) already in the conference.

The exact locking behavior depends on whether or not the Virtual Meeting Room or Virtual Auditorium being used has a Host PIN.

If the service **does not have a Host PIN**:

- Participants are able to join the conference until it is locked.
- When the conference is locked:
 - A conference locked indicator  is displayed.
 - Any further participants who attempt to join the conference (including any Automatically Dialed Participants and manually-invited participants who have been given a role of Guest) are held at the **Waiting for the host** screen. However, any ADPs and manually-invited participants with a role of Host will join the conference immediately.
 - All participants who are already in the conference are notified of any participants who are attempting to join the locked conference, and can [allow the waiting participants to join](#). Notifications take the form of an on-screen message and an audio message/alert for each participant attempting to join.
- If the conference is unlocked, any participants who are still waiting will automatically join the conference.

If the service **has a Host PIN**:

- Host and Guest participants are able to join the conference until it is locked.
- When the conference is locked:
 - A conference locked indicator  is displayed to Host participants.
 - New participants who enter the Host PIN will join the conference immediately — locking does not apply to them.
 - Any new Guest participants (including any Automatically Dialed Participants and manually-invited participants who have been given a role of Guest) are held at the **Waiting for the host** screen.
 - All Host participants who are already in the conference are notified of any Guest participants who are attempting to join the locked conference, and can [allow the waiting Guest participants to join](#). Notifications take the form of an on-screen message and an audio message/alert for each participant attempting to join.
- If the conference is unlocked, any Guest participants who are still waiting will automatically join the conference.

All of the on-screen indicators, messages and the **Waiting for the host** screen can be fully customized via the theme associated with your services.

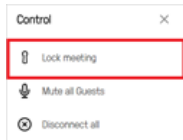
Locking using the Administrator interface

To lock or unlock a conference from the Administrator interface:

1. Log into the Pexip Infinity Administrator interface.
2. Go to **Status > Conferences**.
3. From the **Service name** column, select the conference you want to lock or unlock.
4. At the bottom left of the page, select **Lock conference** or **Unlock conference** as appropriate.

Locking using Infinity Connect

Host participants using Infinity Connect can lock and unlock the conference they are in by going to the side panel, selecting **Control** ● ● ● and then selecting **Lock meeting** or **Unlock meeting** as appropriate:



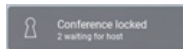
Host participants using Infinity Connect can also use the [commands](#) `/lock` and `/unlock`.

Locking using DTMF

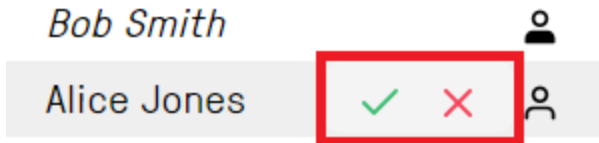
If DTMF controls have been enabled, Host participants using telephones or SIP/H.323 endpoints can lock and unlock the conference using DTMF. The default DTMF entry to do this is *7 but this may have been customized.

Allowing waiting participants to join a locked conference

When a new participant attempts to join a locked conference, all Host participants (on any endpoint) in the conference are notified that a participant is waiting to join. However, only Host participants who are using Infinity Connect can admit individual participants into the conference.



Participants who are waiting to join a locked conference are shown in the **Participant list** with a tick and cross next to their names. To allow these participants to join the conference, select the green tick. If you do not want them to join, select the red cross.



Note that if the Host has joined as presentation and control-only (and there are no other Host participants), the Host is not offered the telephone icons. However, they can use the **Start the meeting** menu option, which will let in all Guest participants.

If a conference is unlocked, all participants who are still waiting will automatically join the conference.

Rejecting a request to join a locked conference

If a Host (who is using Infinity Connect) does not want a waiting participant to join the conference immediately, they have two options:

- To reject the request completely, the Host participant must click on the red cross icon next to the waiting participant's name. The waiting participant's call will be disconnected.
- To leave the participant at the waiting for Host screen, the Host participant should do nothing. The waiting participant will remain at the waiting screen until:
 - a Host participant chooses to let the waiting participant join the conference, or
 - the conference is unlocked (after which the waiting participant will automatically join the conference), or
 - the participant has been waiting for longer than the [specified waiting time](#) (after which the participant will be disconnected)
 - the conference finishes (after which the waiting participant's call will be disconnected).

Muting a participant's audio

There are several ways the audio being sent from one or more participants can be muted by an administrator or a meeting Host:

- [Using the Administrator interface](#)
- [Using Infinity Connect](#)
- [Using DTMF](#)

This is known as being "administratively muted". When a participant's audio has been administratively muted, Pexip Infinity still receives their audio stream but does not add it to the mix being sent to all other participants. This is different to a participant muting their own microphone locally, when the local client or endpoint does not send any audio to Pexip Infinity.

Administrators can also customize Infinity Connect clients so that the microphone is muted locally by default. In these cases, participants are able to subsequently unmute and mute themselves.

Note that low-level, almost imperceptible background noise is added to the audio mix in all conferences. This creates a similar effect to an open mic and gives reassurance that the conference is alive, even if all participants are muted. This background noise is not configurable and cannot be disabled.

Using the Administrator interface

Administrators can use the Pexip Infinity Administrator interface to either mute individual participants, or mute all Guest participants.

Muting an individual participant

To use the Pexip Infinity Administrator interface to mute a participant's audio:

1. Select the participant. You can do this in two ways:
 - Go to **Status > Participants** and select the participant to mute.
 - Go to **Status > Conferences** and select the Virtual Meeting Room or Virtual Auditorium that the participant is in. From the **Participants** tab, select the participant to mute.
2. At the bottom right of the screen, select **Mute**.

Muting all Guest participants

To use the Pexip Infinity Administrator interface to mute the audio from all Guest participants:

1. Go to **Status > Conferences** and select the Virtual Meeting Room or Virtual Auditorium.
2. At the bottom left of the screen, select **Mute all Guests**.

All Guest participants currently in the conference, and any Guest participants who subsequently join the conference, will be muted. Individual Guest participants can still be unmuted and muted by a conference Host or system administrator.

Using Infinity Connect

 You must have Host privileges to use this feature.

Host participants can mute an individual participant, or mute all Guests simultaneously. Note that it does not mute the participant's speakers, so they will still hear all other unmuted participants, but what that muted participant says will not be heard.

When Infinity Connect has been used to mute a participant, the **Audio administratively muted?** column of the Conference status page of the Pexip Infinity Administrator interface will show **YES**.


- Participants will not be notified that they have been muted or unmuted, although Infinity Connect participants will see a muted icon next to themselves in the participant list.
- Participants can mute and unmute themselves using Infinity Connect, but only if they have Host privileges.
- An Infinity Connect user can unmute a participant previously muted by another Infinity Connect user.

Muting an individual participant

To use Infinity Connect to mute or unmute an individual participant's audio:

- From the **Participant** list, select the participant and then select **Mute** or **Unmute**.

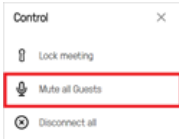
When muted, a  icon is shown next to the participant's name.

 Host participants using Infinity Connect can also use the [commands](#) `/mute [participant]` and `/unmute [participant]`.


Muting all Guest participants

To use Infinity Connect to mute the audio coming from all Guest participants:

- From the top of the side panel, select **Control** ● ● ● and then select **Mute all Guests**.



- i** Host participants using Infinity Connect can also use the [commands](#) `/muteall` and `/unmuteall`.

All Guest participants currently in the conference, and any Guest participants who subsequently join the conference, will be muted (indicated by a  icon next to their name). Individual Guest participants can still be unmuted and muted by a conference Host or system administrator.

Using DTMF

If DTMF controls have been enabled, Host participants can mute and unmute all Guest participants. The default DTMF entry to do this is *5 but this may have been customized. For more information, see [Using a DTMF keypad to control a conference](#).

Manually dialing out to a participant from a conference

The Pexip Infinity platform allows you to dial out to participants from an ongoing conference, on an ad hoc basis. When you dial out to a participant in this way, a call is placed to their endpoint from the Virtual Meeting Room or Virtual Auditorium. If they answer the call, they join the conference as either a Host or Guest, depending on the option that you selected. Participants joining the conference in this way do not go through the IVR screen and do not have to enter a PIN.

The participant could also take the form of a dedicated multimedia stream to enterprise CDN (Content Delivery Network) streaming and recording services such as Wowza, Quickchannel, Qumu, VideoTool, Microsoft Stream and Azure Media Services, and to public streaming services such as YouTube, Facebook and Periscope. Any Pexip conference can be streamed as a live event to an unlimited number of viewers, and can automatically be recorded and stored for later consumption. For more information, see [Streaming and recording a conference](#).

Dialing out from a VMR to an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet is not supported.

Administrators and conference hosts can add participants to a conference. Administrators can do so via the [Administrator interface](#), and hosts can do so via the [Infinity Connect client](#).

- i** You can also configure a Virtual Meeting Room or Virtual Auditorium so that one or more participants are dialed out to automatically whenever a conference starts. See [Automatically dialing out to a participant from a conference](#) for more information.
- i** If your environment includes a PSTN gateway or uses an ITSP (Internet telephony service provider), consider the potential for toll fraud if you have Call Routing Rules that can route calls to the PSTN gateway or ITSP, or if you allow conference participants to dial out to other participants via the PSTN gateway or ITSP. See [PSTN gateways and toll fraud](#) for more information.

When dialing out from a conference, the outbound call bandwidth limit is inherited from the VMR's bandwidth settings. (If a Call Routing Rule is applied, the rule's bandwidth settings are not used.)

Call Routing Rule requirements and controlling dial-out capabilities

The ability to dial out to a participant from a conference is enabled by default. Call Routing Rules may be required when dialing out from a conference to a new participant, depending on the method used:

- Using the Infinity Connect clients or client API:** in most cases, you need to configure appropriate Call Routing Rules to enable end-users using the Infinity Connect clients or the client API to place an outbound call. Rules may not be required for the client API or legacy (webapp1) clients if the **Enable legacy dialout API** setting (**Platform > Global Settings > Connectivity**) is selected — see the [Connectivity options in global settings](#) for more information.

- **Using administrator tools:** when dialing out via the Administrator interface, management API or when using Automatically Dialed Participants (ADPs), you have the option to choose automatic routing (where a rule is required) or to manually specify the call details (where a rule is not required).
- i** This means that when dialing out from an ongoing conference, any calls made via an Infinity Connect client (including RTMP calls to a streaming or recording service) **always** use automatic routing and thus **must** match an appropriate Call Routing Rule for the call to be placed.

Dialing out via the Administrator interface

You can use the Pexip Infinity Administrator interface to dial out to a participant from a Virtual Meeting Room or Virtual Auditorium. If and when the call is answered, that endpoint will join the conference.

To dial out to a participant using the Administrator interface:

1. Select the Virtual Meeting Room or Virtual Auditorium to dial the participant from. You can do this in any of the following ways:
 - Go to **Services > Virtual Meeting Rooms** and select the name of the Virtual Meeting Room.
 - Go to **Services > Virtual Auditoriums** and select the name of the Virtual Auditorium.
 - If the conference that you want to dial out from is already in progress, go to **Status > Conferences** and select the required Virtual Meeting Room or Virtual Auditorium.
2. At the bottom left of the screen, select **Dial out to participant**.
3. Complete the following fields:

Field	Description
System location	<p>Select the system location from which the call will be placed. If there is more than one Conferencing Node in that location, Pexip Infinity will choose the most appropriate.</p> <p>The system location determines which H.323 gatekeeper or SIP proxy to use to route the call.</p> <p>i You should not select a Pexip Smart Scale location.</p>
Service alias	<p>This lists all the aliases that have been configured for the selected Virtual Meeting Room or Virtual Auditorium. The participant will see the incoming call as coming from the selected alias.</p>
Participant alias	<p>The alias of the endpoint/participant that you want to dial.</p>
Route this call	<p>Select how to route the call:</p> <ul style="list-style-type: none"> ◦ Manually: uses the requested Protocol and the defaults for the specified System location. ◦ Automatically: routes the call according to the configured Call Routing Rules. This means that the dialed alias must match an outgoing Call Routing Rule for the call to be placed (using the protocols, outgoing location and call control systems etc. as configured for that rule). <p>Default: Manually.</p>
Participant display name	<p>An optional user-facing display name for this participant, which may be used in participant lists and as the overlaid participant name (if enabled).</p> <p>If this name is not specified then the Participant alias is used as the display name instead.</p>
Protocol	<p>The signaling protocol to use when dialing the participant. Select either SIP, H.323, or if the endpoint is a Skype for Business / Lync client, select Lync / Skype for Business (MS-SIP). The RTMP protocol is typically used when adding a streaming participant. Note that if the call is to a registered device, Pexip Infinity will instead use the protocol that the device used to make the registration.</p> <p>This field only applies when Route this call is set to Manually.</p>
Call capability	<p>Allows you to limit the media content of the call. The participant being called will not be able to escalate beyond the selected capability. For more information, see Controlling media capability.</p> <p>Default: Main video + presentation.</p> <p>This field only applies when Route this call is set to Manually.</p>

Field	Description
Role	Select whether you want the participant to join the conference as a Host or Guest .
DTMF sequence	<p>An optional DTMF sequence to be transmitted after the call to the dialed participant starts.</p> <p>A DTMF sequence can include: the digits 0-9, "*" (asterisk), "#" (hash) or "," (comma).</p> <p>The DTMF tones are sent 3 seconds after the call connects, one at a time, every 0.5 seconds. A comma is a special digit that represents a 2 second pause (multiple commas can be used if a longer pause is needed).</p> <p>For example, if you are dialing an audio bridge and want to enter conference number 777 followed by #, pause for six seconds and then supply conference PIN 1234 followed by #, you would configure 777#,,,1234# as your DTMF sequence.</p>
Streaming	Identifies the dialed participant as a streaming or recording device. When a conference participant is flagged as a streaming/recording participant, it is treated as a receive-only participant and is not included in the video stage layout seen by other participants. See Streaming and recording a conference for more information.
Dual stream (presentation) URL	<p>When adding a dual streaming RTMP participant, this specifies the RTMP URL for the second (presentation) stream.</p> <p>Leave this field blank when adding a single streaming participant.</p>
Keep conference alive	<p>Determines whether the conference will continue when all other participants have disconnected:</p> <ul style="list-style-type: none"> ○ Yes: the conference will continue to run until this participant has disconnected (applies to Hosts only). ○ If multiple: the conference will continue to run as long as there are two or more If multiple participants and at least one of them is a Host. ○ No: the conference will be terminated automatically if this is the only remaining participant. <p>Default: Yes.</p> <p>For streaming participants, we recommend that this option is set to No.</p> <p>For more information, see Automatically ending a conference.</p>

4. Select Dial out to participant.

A message **Initiated dial out to participant** appears at the top of the screen.

To confirm whether the participant has joined the conference you can go to **Status > Conferences**, select the conference, and then select the **Participants** tab. The new participant should appear in the list.

Dialing out via the Infinity Connect client

If you have Host privileges, you can use the Infinity Connect client to dial out to participants from the conference you are in. A call is placed to the participant and if they answer the call they will join the conference directly (they do not go through an IVR screen or have to enter a PIN).

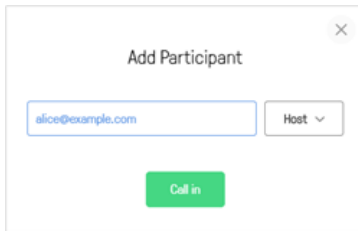
i Automatic routing is used when an Infinity Connect client adds a new participant to a conference. This means that the dialed alias must match an appropriate Call Routing Rule that applies to **Outgoing calls from a conference** for the call to be placed (using the protocols and call control systems etc. as configured for that rule).

To dial out to a participant using Infinity Connect, from within a conference:

1. From the toolbar at the bottom of the screen, select **Add participant**:



2. At the prompt, enter the address of the person you want to dial:



3. Select whether you want the participant to have **Host** or **Guest** privileges.
4. Select **Call in**.

The call is placed from the conference to the participant and they appear in the participant list with a green line under their name while their endpoint is ringing. If and when the participant answers the call they will join the conference; if they do not answer, or do not accept the call, they will disappear from the participant list.

For **legacy** clients, there is the option to select a protocol of *Automatic* (the default), *SIP*, *H.323*, *Lync/Skype* or *RTMP*. To successfully place calls via the Automatic protocol option, suitable Call Routing Rules must be configured. To enable calls to be placed via the other protocols you must select **Enable legacy dialout API** (via **Platform > Global Settings > Connectivity**).

Disconnecting participants from a conference

There are several ways to disconnect a participant from a conference: [Using the Administrator interface](#), [Using Infinity Connect](#) and [Using DTMF](#). Each of these options is described below.

Using the Administrator interface

Disconnecting a single participant

To use the Pexip Infinity Administrator interface to disconnect a single participant from a conference:

1. Select the participant. You can do this in two ways:
 - Go to **Status > Participants** and select the participant to disconnect.
 - Go to **Status > Conferences** and select the Virtual Meeting Room or Virtual Auditorium that the participant is in. From the **Participants** tab, select the participant to disconnect.
2. At the bottom right of the screen, select **Disconnect**.

Disconnecting all participants from a conference

To use the Pexip Infinity Administrator interface to disconnect all of the participants from a conference:

1. Go to **Status > Conferences** and select the Virtual Meeting Room or Virtual Auditorium that the participants are in.
2. At the bottom right of the screen, select **Disconnect all**.
3. Select **Yes, I'm sure** to confirm.

All participants will be disconnected and the conference will no longer be active.

Using Infinity Connect

i You must be a Host participant in the conference to use this feature.

Disconnecting a single participant

To use an Infinity Connect client to disconnect a participant:

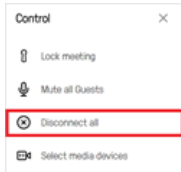
- From the participant list, select the participant's name and then select **Disconnect**.

i Host participants using Infinity Connect can also use the `command /disconnect [participant]`.

Disconnecting all participants

To use an Infinity Connect client to disconnect all participants, including yourself:

- From the top of the side panel, select **Control** ● ● ● and then select **Disconnect all**.



i Host participants using Infinity Connect can also use the [command](#) `/disconnectall` .

Using DTMF

If DTMF controls have been enabled, Host participants can terminate the conference by disconnecting all participants, including themselves. The default DTMF entry to do this is `##` but this may have been customized. For more information, see [Using a DTMF keypad to control a conference](#).

Transferring a participant to another conference

You can transfer a participant from one conference to another conference (or to another device or system via the Pexip Distributed Gateway). There are several ways to do this: [Using the Administrator interface](#), [Using Infinity Connect clients](#), or [Using the APIs](#).

After the transfer has been successfully initiated, the transferred participant will be removed from the original conference. Should the transfer then fail (for example, if it is made to a device that rejects the call), the transferred participant will be disconnected i.e. it will not rejoin the original conference.

Note that when a Skype for Business / Lync participant is transferred, they will appear as audio-only and will need to escalate to video.

Using the Administrator interface

To use the Pexip Infinity Administrator interface to transfer a single participant from one conference to another conference:

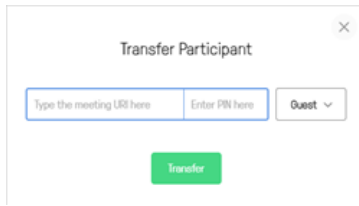
1. Select the participant. You can do this in two ways:
 - Go to **Status > Participants** and select the participant to transfer.
 - Go to **Status > Conferences** and select the Virtual Meeting Room or Virtual Auditorium that the participant is currently in. From the **Participants** tab, select the participant to transfer.
2. At the bottom right of the screen, select **Transfer**.
The **Transfer Participant** dialog opens.
3. Enter the **Target conference alias**.
This can be any alias associated with a Virtual Meeting Room, Virtual Auditorium or Call Routing Rule.
4. Select the **Role** the participant will have in the new conference.
5. Select **Transfer**.
If the transfer is successful, the participant list will be updated to show the participant in the new conference (it may also temporarily show the participant still connected to the previous conference).

Using Infinity Connect clients

To use the Infinity Connect client to transfer a participant from one conference to another:

1. Join the conference as a Host.
2. From the **Participant list**, select the participant and then select **Transfer Participant**.
Enter the alias of the conference you wish to transfer the participant to, the PIN (if applicable) and whether they should join as a Guest or Host, and then select **Transfer**.

You can transfer any participant, including yourself.

A screenshot of a web-based dialog box titled "Transfer Participant". It has a close button (X) in the top right corner. Inside the dialog, there are two input fields: "Type the meeting URI here" and "Enter PIN here". To the right of these fields is a dropdown menu labeled "Guest". Below the input fields is a green button labeled "Transfer".

Using the APIs

Third-party applications can use the management API and the client APIs to transfer participants to another conference.

For more information, see [Management command API](#) and [Using the Pexip client APIs](#).

Using a DTMF keypad to control a conference

Host participants using telephones or SIP/H.323 endpoints that support DTMF can control aspects of a conference by using their keypad.

Default DTMF controls

The default DTMF controls that can be used within a conference (Virtual Meeting Room or Virtual Auditorium) are:

DTMF digits	Control
*7	Toggle conference lock and unlock
*5	Toggle mute and unmute all Guests
##	Terminate the conference (disconnect all participants including yourself)

Note that with calls made via the Infinity Gateway, any DTMF signals are forwarded to the other party.

Changing DTMF controls

The DTMF digits for each control can be changed on a per-theme basis by editing the [themeconfig.json file](#).

Each control must be two DTMF digits long, and all the commands must be different. When deciding the digits to use for each control, you should ensure there is no risk of overlap in situations where one of the digits is not successfully received.

For example, if you used:

- 57 to lock the conference
- 75 to mute all Guests
- 77 to end the conference

there is a risk that a user could enter 5775 to lock and mute the conference, but due to packet loss the initial digit was not received. In this case the string that is received would be 775, which would be interpreted as 77 and thus a command to terminate the conference.

Disabling DTMF controls

DTMF controls are enabled by default. They can be disabled and enabled on a per-theme basis by editing the [themeconfig.json file](#).

Customizing with themes

Customizing conference images and voice prompts using themes

Themes allow you to change the voice prompts and images provided to participants when they use a Pexip Infinity service (Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service) or use the Infinity Gateway to make a person-to-person call or to join an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet. You might change the theme if, for example, you want to use your company's own logo, color scheme or terminology on the screens displayed to conference participants, or you want to change the language used in the voice prompts.

You have full flexibility when applying themes to services. You apply specific themes to individual services and Call Routing Rules, and you can nominate a global [default theme](#) to use for any services that have not had a specific theme applied to them.

Each conference theme is made up of a set of audio and image files and configuration settings. You can replace any or all of these files with your own customized sounds, images or settings in order to [create a new theme](#) and [apply a theme to a specific service](#).

Pexip Infinity ships with its own [Base theme](#), which cannot be edited. Files and configuration settings from the Base theme are used when no other theme has been selected, or when the selected theme does not contain a specific file or setting. Pexip Infinity also ships with a number of [preconfigured themes](#) — these include alternative themes that contain entry and exit tones and localized audio files — making it easier for you to select and create themes suitable for your deployment.

Pexip branding portal

We recommend using the Pexip branding portal (<https://brandingportal.pexip.com>) to customize your themes. The portal provides an easy way to change the text, images and so on that are used in themes, and it generates a ZIP file that you can upload via the Administrator interface. Note that you cannot change the audio files via the branding portal.

You can also [manually configure](#) your own themes as described in this documentation if you have more advanced customization requirements, or want to use your own audio files.

If you use the Infinity Connect clients to access your services, you can also use the branding portal to customize the look and feel of those clients. For more information, see [Customizing the Infinity Connect clients](#).

How do I know which files and configuration settings will be used in a particular VMR?

When you create a new customized theme, you don't have to upload a complete set of files and configuration settings — you can just upload the files and settings you want to use specifically for that theme, to override the default behavior. When a user accesses a service (such as a VMR) that has that theme applied, Pexip Infinity presents them with the sounds and images from the files included in that theme. If the customized theme does not include a specific file or setting, or the service does not have a theme applied, then Pexip Infinity uses the relevant file or setting from the theme that has been selected as the [default theme](#). If the default theme does not include the specific file or setting either, then the file or setting from the [Base theme](#) is used.

For example, this diagram (right) shows how some files and settings in the Base theme (the `conf-waitforleader_48kHz_mono.wav` audio file, and the `streaming_indicator_text`, `disable_streaming_indicator` and `disable_watermark_icon` settings in the `themeconfig.json` file, all shown faded out) are overridden by the equivalent files or settings in the default theme (the `disable_streaming_indicator` setting in this example) and any service-level theme (the `conf-waitforleader_48kHz_mono.wav` file, and the `streaming_indicator_text` and `disable_watermark_icon` settings).

Note that, in this example, even though the default theme includes the `disable_watermark_icon` setting, that setting is itself overridden by the service level theme; however, if a service did not have that service-level theme applied, then the default theme setting of `disable_watermark_icon: true` would apply to that service.

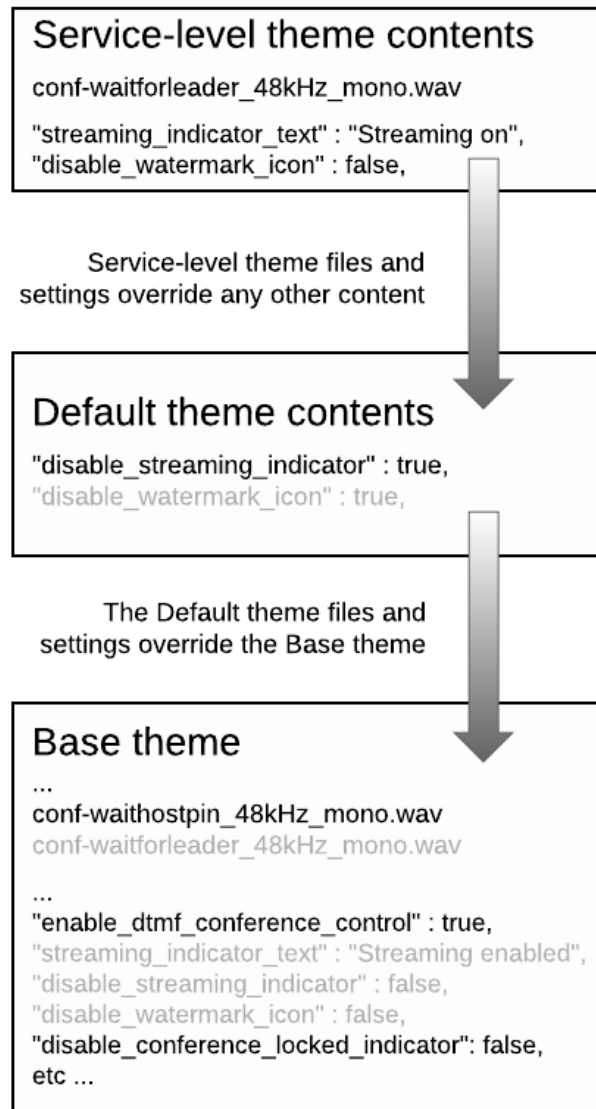
If a theme is only assigned to Call Routing Rules, you only need to customize the relevant subset of the image files (see [Themes used by Call Routing Rules \(gateway calls\)](#) for more information).

If you use the Pexip branding portal to create your theme it will automatically include just the relevant settings and files required to override the default behavior.

New and legacy style themes

Version 18 of Pexip Infinity introduced a new way to specify the content and layout of your own customized themes that is more efficient than the previous style themes, and offers more flexibility when customizing them for your own requirements.

Legacy style themes (identified as "version 1" themes) and new style themes (identified as "version 2" themes) can both be used on the same platform, with legacy style themes applied to some services and new style themes applied to other services if required. The way in which themes are uploaded to Pexip Infinity and applied to services is the same for both new and legacy style themes. The new style base theme is used by default if you have not applied any customized themes to your services.



Both types of themes behave in exactly the same manner as each other, and look almost the same by default (some of the PIN-entry splash screen graphics used when joining a conference are different). The audio prompts are the same in new and legacy style themes.

The main difference between how legacy and new style themes work and are configured/customized, is that legacy style themes use a JPG file containing a combined background image, graphics and text for each of the splash screens used when joining a conference (such as the Welcome screen and PIN entry screens). The new style themes separate out these elements, allowing you to specify the individual background image, graphics and text elements that are used on each screen, and control where each of those elements is positioned (via new configuration options in the theme's `themeconfig.json` file). See the section on [splash screens](#) for more information.

Controlling whether new or legacy theme files are used

A configuration switch in the theme's `themeconfig.json` file is used to control whether the theme is a new or legacy style — if it contains `"theme_version": 2` then new style themes are used. This switch has to be present in a `themeconfig.json` file at the lowest level of the theme file hierarchy where a theme has been applied i.e. if you have applied a theme to a specific service then, for new style themes to be used, that theme must contain a `themeconfig.json` file that includes the `"theme_version": 2` switch.

- Therefore, even if you only want to customize a single image or sound file, you must include in your ZIP upload a `themeconfig.json` file that contains a `"theme_version": 2` switch, if you want to use new-style themes.

The flow chart shows how the inclusion or not of `"theme_version": 2` controls whether a new or legacy-style theme is displayed.

The Pexip branding portal can be used to customize new and legacy-style themes, and any themes created via the branding portal automatically include the relevant new/legacy switches in the theme's `themeconfig.json` file when required.

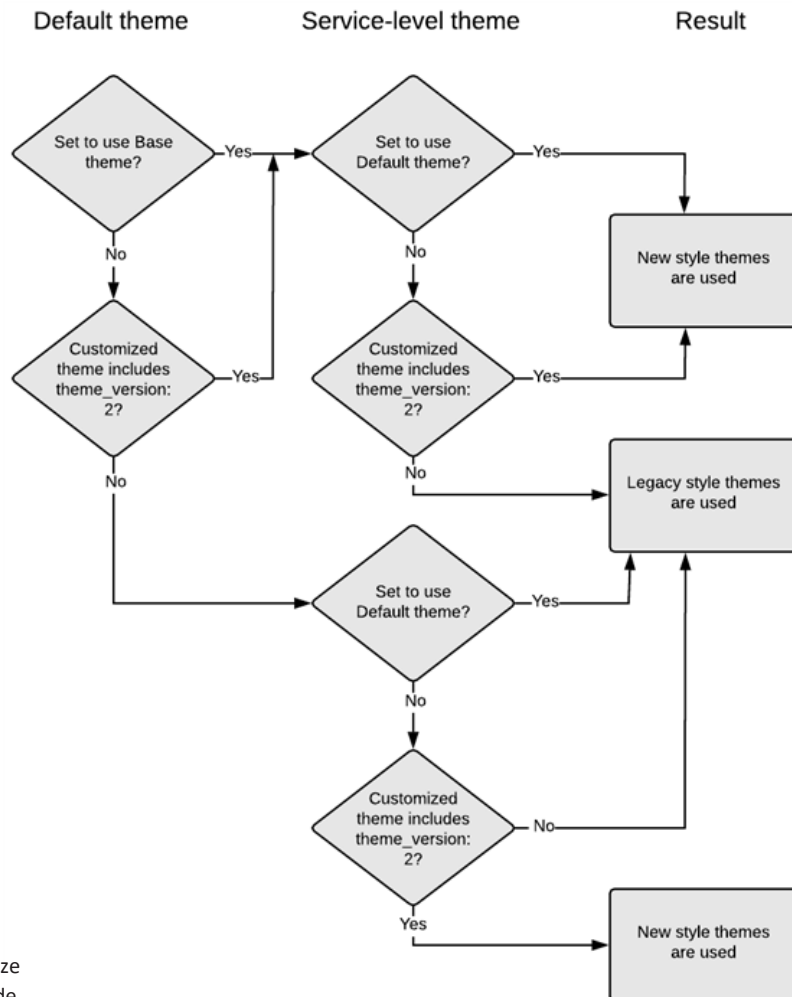
Creating and applying themes to conferences

This section describes how to [create](#) and [edit](#) your own themes, change the [default theme](#), and how to [apply a theme to a specific service](#) or see which themes you are [currently using](#).

Legacy style themes (identified as "version 1" themes) and new style themes (identified as "version 2" themes) can both be used on the same platform, with legacy style themes applied to some services and new style themes applied to other services if required. The way in which themes are uploaded to Pexip Infinity and applied to services is the same for both new and legacy style themes. The new style base theme is used by default if you have not applied any customized themes to your services. See [New and legacy style themes](#) for more information.

Creating a new theme

You can create a new theme by uploading a ZIP file containing the files or configuration settings you want to use instead of the default theme. The ZIP file does not need to contain the complete set of files or settings, just the files or settings that you want to use in place of, or as an extension to, those from the default theme.



We recommend using the Pexip branding portal (<https://brandingportal.pexip.com>) to customize your themes. The portal provides an easy way to change the text, images and so on that are used in themes, and it generates a ZIP file that you can upload via the Administrator interface. Note that you cannot change the audio files via the branding portal.

To create and upload a new theme:


1. Obtain your theme ZIP file:
 - If you have used the Pexip branding portal to customize your theme:
 - i. Build and download the theme (Splash Screens) from the branding portal.
This creates an `ivr.zip` file.
 - If you are manually creating your own customized theme:
 - i. Ensure that any new sound and image files and the `themeconfig.json` file meet the specified [file requirements for themes](#).
 - ii. Save all of the files together in a new folder. You can name the folder whatever you like, but within the folder, each file must be saved with the same file name and extension as the default file being replaced. If you are adding extra background files or SVG icon files then you can use any suitable name for those additional files.
 - iii. Create a .ZIP file of the new folder.
2. Go to **Services > Themes** and select **Add theme**.
3. In the **Name** field, enter the name of the new theme.
4. In the **Theme** field, select **Choose file** and go to the ZIP file you have just downloaded/created.
5. In the **Services** and **Call Routing Rule** sections, select the services (Virtual Meeting Rooms, Virtual Receptions, Call Routing Rules and so on) that will use this new theme.
Note that if you have more than 10,000 services configured, you cannot select the associated services via this page. Instead you must use another method to assign the theme to a service, such as [editing the individual service](#), using the configuration API, or by using the [bulk VMR provisioning](#) facility.
6. Select **Save**.
If any errors are reported when trying to upload your own manually-created ZIP file, such as "Invalid file in theme: themeconfig.json", you must fix the errors and then try to upload the ZIP file again. Any errors in the themeconfig.json are typically due to invalid JSON structures, such as missing commas between any key value pairs, or having a comma after the last key value pair in any object. Graphics file related errors are typically caused by incorrect image resolutions or using an invalid filename.
7. Wait for the new theme to be replicated out to all Conferencing Nodes (typically after approximately one minute).

You can now test the theme by dialing one of the services to which it has been applied.

Setting the default theme

You can specify a theme to be used by default for all services that have not had a specific theme applied. The default theme does not need to contain a complete set of files or configuration settings, just those that you want to use in place of files from the [Base theme](#). It can be either a legacy or new style theme. If you use the Pexip branding portal to create your theme, it will automatically include just the relevant settings and files.

If you do not select a default theme, the entire Base theme is used as the default. To change the default theme:

1. Go to **Platform > Global Settings > Service Configuration**.
2. From the **Default theme** drop-down, select the theme to use for services where no specific theme has been selected. If you have not already uploaded the theme you want to use, you can do so from here by clicking on the  icon and following the instructions in [creating a theme](#).
3. Select **Save**.

Editing an existing theme

You can edit an existing theme by uploading a new ZIP file containing the new set of files. The ZIP file does not need to contain the [complete set of files or settings](#), just the files or settings that you want to use in place of, or as an extension to, those from the default theme.

Note that when you upload a ZIP file for an existing theme, all of the previously uploaded files for that theme are deleted and replaced with files from the new ZIP file. This means that if you have an existing theme containing **File A** and you want to add **File B** to the

theme, you need to upload a ZIP file containing both **File A** and **File B**. If you use the Pexip branding portal to edit and then rebuild your theme it will automatically include all of the relevant settings and files.

To modify an existing theme:

1. If you originally used the Pexip branding portal to create your theme:
 - a. Go to the branding portal <https://brandingportal.pexip.com>.
 - b. Select and then edit your existing theme as required.
 - c. Build and download the updated theme from the branding portal as a new ZIP file.
2. If you are manually customizing your theme:
 - a. If you want to retain files from the existing theme, first [download](#) the current theme.
 - b. Make your required changes to the theme files, ensuring that any new sound and image files and the `themeconfig.json` file meet the specified [file requirements for themes](#).
 - c. Save all of the new and modified files, along with any other existing files, together in a new folder. You can name the folder whatever you like, but within the folder, each file must be saved with the same file name and extension as the default file being replaced.
 - d. Create a ZIP file of the new folder.
3. Go to **Services > Themes** and select the theme you want to update.
4. In the **Theme** field, select **Choose file** and go to the ZIP file you have just downloaded/created.
5. Select **Save**.

If any errors are reported when trying to upload your own manually-created ZIP file, such as "Invalid file in theme: themeconfig.json", you must fix the errors and then try to upload the ZIP file again. Any errors in the `themeconfig.json` are typically due to invalid JSON structures, such as missing commas between any key value pairs, or having a comma after the last key value pair in any object. Graphics file related errors are typically caused by incorrect image resolutions or using an invalid filename.

6. Wait for the updated theme to be replicated out to all Conferencing Nodes (typically after approximately one minute).

You can now test the theme by dialing one of the services to which it has been applied.

Downloading an existing theme

You can download a theme to view the files it contains, or to manually [edit](#) it. To download a theme:

1. Go to **Services > Themes**.
2. Either:
 - Select **Download Base theme** to download a copy of the default set of files that are shipped with Pexip Infinity.
 - Select **Download Legacy Base theme** to download the base files that applied to legacy-style themes.
 - Select an existing theme and from the **Change Theme** page select **Download theme** to download a copy of the files that make up that customized theme.

A file with a .ZIP extension will be downloaded.

Changing which themes are associated with which services

There are many ways in which you can change the theme that is associated with a service or Call Routing Rule. You can do this:

- via the **Themes** page (go to **Services > Themes**, select the new theme you want to use, then select the services you want to apply it to) — you cannot use this method if you have more than 10,000 services configured
- by editing the individual service (go to **Services**, select the service, then select the new theme from the **Theme** drop-down menu) or Call Routing Rule (**Services > Call Routing**)
- via the configuration API
- by using the [bulk VMR provisioning](#) facility.

Viewing the themes currently in use

To find out which themes are currently in use in your deployment, go to **Services > Themes**. The **Service count** column shows you how many services each theme is associated with.

Themes are also shown on the overview page for each type of service (go to **Services** and then select the service). Here you can sort the list by **Theme**.

Base theme and other preconfigured themes

Pexip Infinity ships with its own Base theme. Files and configuration settings from the Base theme are used when no other theme has been selected, or the selected theme does not contain the required file or setting.

You cannot change the contents of the Base theme, but you can [create your own themes](#) by uploading customized versions of one or more of the Base files.

When a user accesses a service that has a customized theme applied, Pexip Infinity presents them with the sounds and images from the files included in that theme. If the customized theme does not include a specific file, or the service does not have a theme applied, then Pexip Infinity uses the relevant file from the [default theme](#). If the default theme does not include the specific file either, then the file from the Base theme is used.

The same rules apply when modifying the configuration settings in the `themeconfig.json` file. If any values **within** this file are not specified, Pexip Infinity will use the values from the Base theme's `themeconfig.json` file.

In addition to the files and settings included within the Base theme, there are some advanced settings (such as controlling where screen text is positioned) that are used by default but are not contained with the Base theme's `themeconfig.json` file, but they can be overridden by adding them to your own customized `themeconfig.json` file. See [Splash screen elements \(to control the size and position of text/graphics\)](#) for full information about what can be configured, and [Rules and requirements for customized themes](#) to see the default positions, offsets and sizes used by each splash screen.

Pexip Infinity also ships with a number of [preconfigured themes](#) which are similar to the Base theme, but contain alternative audio files with different conference entry and exit tones, and alternative references to the "#" key (as the "hash key" or as the "pound key").

Preconfigured themes

Pexip Infinity ships with a number of preconfigured themes. These themes can be copied and edited, making it easier for you to select and create themes suitable for your deployment.

The preconfigured themes are identical to the Base theme with the following exceptions:

- some include [entry and exit tones](#) (the Base theme contains "empty" tones files)
- some refer to the "#" key as the "hash key" (the Base theme refers to this as the "pound key").

The files that may differ from those in the Base theme are:

- **Entry tone:** `conf-participant_entry_tone_48kHz_mono.wav`
- **Exit tone:** `conf-participant_exit_tone_48kHz_mono.wav`
- **"#" key reference:** `2sd-number-pound-key_48kHz_mono.wav`, `conf-getpin_pound-key_48kHz_mono.wav`, `conf-waithostpin_pound-key_48kHz_mono.wav`

The content of the above files for each of the preconfigured themes is as follows:

Theme	Entry tone	Exit tone	"#" key references
Pexip theme (English_US) *	<empty file>	<empty file>	"...the pound key"
Pexip theme (English_UK)	<empty file>	<empty file>	"...the hash key"
Pexip theme (English_US) with entry tones	A high tone followed by a low tone	A low tone followed by a high tone	"...the pound key"
Pexip theme (English_UK) with entry tones	A high tone followed by a low tone	A low tone followed by a high tone	"...the hash key"

* This theme is identical to the Base theme, although unlike the Base theme it can be edited.

Rules and requirements for customized themes

We recommend using the Pexip branding portal (<https://brandingportal.pexip.com>) to customize your themes. The portal provides an easy way to change the text, images and so on that are used in themes, and it generates a ZIP file that you can upload via the Administrator interface. Note that you cannot change the audio files via the branding portal.

If you want to manually customize your own themes, or have more advanced customization requirements, the following sections list the complete set of [audio](#), [configuration](#) and [image](#) files that can be used to configure each theme. They include the requirements for each of those files, including how to manage [video watermarking](#), and information on which files are relevant to themes used by [Call Routing Rules](#).

Theme ZIP file

When [creating or editing a theme](#), you upload a single ZIP file containing your own customized version of one or more of the files described here. The ZIP file does not need to contain the complete set of files or settings, just the files or settings that you want to use in place of, or as an extension to, those from the default [Base theme](#).

If, for example, you are replacing one of the default background JPGs, icon SVG files or audio files, then you should ensure that you use the same file name as the file you are replacing. If you are using additional background or icon files that you are applying to a subset of screens, then you must also include those new files in your ZIP file.

If `"theme_version": 2` is not present in a `themeconfig.json` file at the lowest level of the theme file hierarchy then it is assumed to be a legacy-style theme (see [How do I know which files and configuration settings will be used in a particular VMR?](#) for more information).

i Therefore, even if you only want to customize a single image or sound file, you must include in your ZIP upload a `themeconfig.json` file that contains a `"theme_version": 2` switch, if you want to use new-style themes.

If any errors are reported when trying to upload your own manually-created ZIP file, such as "Invalid file in theme: themeconfig.json", you must fix the errors and then try to upload the ZIP file again. Any errors in the `themeconfig.json` are typically due to invalid JSON structures, such as missing commas between any key value pairs, or having a comma after the last key value pair in any object. Graphics file related errors are typically caused by incorrect image resolutions or using an invalid filename.

Audio (sound) files

You can replace any of the audio files by using your own version of the file in your customized theme. You must use the same file name as the file you are replacing. All audio files must be:

- .WAV format
- RIFF (little-endian) data, WAVE audio, Microsoft PCM, 16 bit, mono 48000 Hz

Points to note:

- **mono** and **48000Hz** are essential - audio files that do not meet these requirements will fail to upload.
- The volume level of the audio recording is important - use the default Pexip Infinity prompts as a guide.
- Some endpoints may take a few seconds after a call connects before they are able to receive audio. For this reason, we have included a 2-second pause at the start of any audio files that may be played when a user first connects to Pexip Infinity. We recommend that you include a similar pause; use the default Pexip Infinity files as a guide.

i If you have used the Pexip branding portal to generate your theme ZIP file and you also want to customize the audio prompts (you cannot currently customize the audio files via the branding portal), then you must add your customized audio (wav) files to the ZIP file generated by the branding portal.

The following table lists the default audio files and their content, which is contained within the Base theme:

Default audio files

These are the default audio files in the Base theme:

File name	Content in Base theme
2sd-invalid-number-three-times-disconnect_48kHz_mono.wav	"You have entered an invalid number three times. I will now disconnect the call."
2sd-not-entered-valid-number-disconnect-call_48kHz_mono.wav	"You have not entered a valid number. I will now disconnect the call."
2sd-number-not-valid-try-again_48kHz_mono.wav	"That number is not valid. Please try again."

File name	Content in Base theme
2sd-number-pound-key_48kHz_mono.wav	"Please enter the number you wish to connect to, followed by the pound key." +
2sd-please-hold-connect-you_48kHz_mono.wav	"Please hold while I try to connect you."
conf-call-will-be-disconnected_48kHz_mono.wav	"Your call will be disconnected."
conf-capacity_exceeded_48kHz_mono.wav	"The conferencing system capacity has been exceeded."
conf-getpin_48kHz_mono.wav	"Please enter the conference PIN number."
conf-getpin_pound-key_48kHz_mono.wav	"Please enter the conference PIN number, followed by the pound key." +
conf-insufficient_licenses_48kHz_mono.wav	"There are insufficient conferencing system licenses available."
conf-invalid_license_48kHz_mono.wav	"The conferencing system license is invalid."
conf-invalidpin_48kHz_mono.wav	"The PIN is invalid for this conference."
conf-leaderhasleft_48kHz_mono.wav	"The Host has left the conference. The conference is about to end."
conf-participant_entry_tone_48kHz_mono.wav	<empty file> (For more information, see Playing notification tones when participants join or leave a conference.)
conf-participant_exit_tone_48kHz_mono.wav	<empty file>
conf-participant_is_in_lobby_48kHz_mono.wav ‡	"Welcome to the lobby. Please wait and your meeting host will admit you soon."
conf-participant_locked_out_48kHz_mono.wav	Three knocks. (For more information, see Locking a conference and allowing participants to join a locked conference.)
conf-placeintoconf_48kHz_mono.wav	"Welcome to the conference."
conf-public_streaming_started_48kHz_mono.wav ‡‡ ◊	"This call is being streamed publicly."
conf-public_streaming_stopped_48kHz_mono.wav ‡‡ ◊	"Public streaming of this call has stopped."
conf-recording_started_48kHz_mono.wav ††	"This call is being recorded."
conf-recording_stopped_48kHz_mono.wav ††	"Recording of this call has stopped."
conf-streaming_started_48kHz_mono.wav ‡‡	"This call is being streamed."
conf-streaming_stopped_48kHz_mono.wav ‡‡	"Streaming of this call has stopped."
conf-test_call_48kHz_mono.wav	"Let's test your video and audio. Count out loud from one to three, now." (For more information, see Configuring the Test Call Service.)
conf-test_call_audio_only_48kHz_mono.wav	"Let's test your audio settings. Count out loud from one to three, now."
conf-test_call_disconnect_48kHz_mono.wav	"If you have technical issues, check your settings or contact your administrator."
conf-transcribing_started_48kHz_mono.wav ‡	"This call is being transcribed."
conf-transcribing_stopped_48kHz_mono.wav ‡	"Transcribing of this call has stopped."
conf-waitforleader_48kHz_mono.wav	"Waiting for the conference Host to join."

File name	Content in Base theme
conf-waithostpin_48kHz_mono.wav	"Waiting for the conference Host to join. If you are the conference Host, please enter the conference PIN number now."
conf-waithostpin_pound-key_48kHz_mono.wav	"Waiting for the conference Host to join. If you are the conference Host, please enter the conference PIN number followed by the pound key." †
conf-you_are_the_only_participant_48kHz_mono.wav	"You are the only participant in the conference."

† Alternative recordings using the term "hash key" are also available - see [Preconfigured themes](#).

†† These audio files are only played to callers who are gatewayed via Pexip Infinity into a Google Meet or Microsoft Teams conference.

‡ These audio files are only played to callers who are gatewayed via Pexip Infinity into a Microsoft Teams conference.

‡‡ These audio files are only played to callers who are gatewayed via Pexip Infinity into a Google Meet conference.

◊ Note that Google Meet public streaming has not been launched by Google yet.

Theme configuration file (themeconfig.json)

The **themeconfig.json** file is the theme's primary configuration file. It is a JSON dictionary that specifies the background and layout of the splash screens used when joining a conference (such as the Welcome screen and PIN entry screens), specifies which image files are used, and controls some of the features and overlays that are used within a conference. It does not control any of the audio (sound) elements of the theme.

i Whenever you include a **themeconfig.json** file in your customized theme, it must contain: `"theme_version": 2`. If it does not contain this setting, then it is assumed to be a legacy-style theme.

If any settings **within** your customized **themeconfig.json** file are not specified, Pexip Infinity will simply use the equivalent setting from the Base theme or use default behavior. For example, if the only thing you want to customize from the default settings is to disable watermarks, then your **themeconfig.json** file only needs to contain the **disable_watermark_icon** setting (in addition to `"theme_version": 2`). See [How do I know which files and configuration settings will be used in a particular VMR?](#) for more information.

Here is the default (Base theme) contents of **themeconfig.json**:

```
{
  "dtmf_conference_control_commands" : {
    "**7" : "toggle_lock",
    "**5" : "toggle_guest_mute",
    "##" : "end_conference"
  },
  "plus_n_indicator_text_color" : "0xFFFFFFFF",
  "recording_indicator_text" : "Recording",
  "conference_locked_indicator_text": "Conference locked",
  "test_call_service_media_delay" : 2,
  "disable_streaming_indicator" : false,
  "disable_conference_locked_indicator": false,
  "streaming_indicator_text" : "Streaming enabled",
  "public_streaming_indicator_text": "Public streaming enabled",
  "enable_dtmf_conference_control" : true,
  "disable_watermark_icon" : false,
  "plus_n_indicator_bg_color" : "0x323232",
  "test_call_service_disconnect_timeout" : 10,
  "theme_version": 2,
  "conference_unlocked_indicator_text": "Conference unlocked",
  "conference_locked_indicator_n_waiting_text": "{number_of_waiting_participants} waiting for host",
  "transcribing_indicator_text": "Transcribing"
}
```

The default settings within the Base **themeconfig.json** file are described below. Note that the Base file does not contain any [text overlay](#) or [splash screen layout](#) controls, but you can add your own controls to your customized file — as described in this topic — to override the default behavior.

The **themeconfig.json** file must be UTF-8 encoded, therefore when editing the file you should use a UTF-8 capable editor. We recommend that you do **not** use Notepad on Windows computers.

Description and default values of themeconfig.json settings

All of the colors in these controls are specified using RGB hexadecimal notation (in the format 0xnnnnnn).

Name	Description	Value in Base theme
dtmf_ conference_ control_ commands	<p>If DTMF controls are enabled, this section specifies the DTMF digits used for each control.</p> <p>Current controls are:</p> <ul style="list-style-type: none"> <code>toggle_lock (*7)</code>: toggles the locked/unlocked state of the conference <code>toggle_guest_mute (*5)</code>: toggles the muted/unmuted status of all Guest participants <code>end_conference (##)</code>: terminates the conference <p>When changing the DTMF controls, you must only edit the digits inside the first pair of quotes on each line (i.e. <code>*7</code>, <code>*5</code> and <code>##</code>). Editing anything else will disable DTMF commands.</p> <p>Each control must be two DTMF digits long, and all the commands must be different. See Changing DTMF controls for guidance on best practice.</p>	<pre>"*7" : "toggle_lock" "*5" : "toggle_guest_mute" "##" : "end_conference"</pre>
plus_n_ indicator_text_ color †	The color of the text on the thumbnail that shows how many additional participants are in the conference. This is used in conjunction with <code>plus_n_indicator_bg_color</code> .	0xFFFFFF (white)
recording_ indicator_text	The text that is associated with the recording indicator and that is temporarily displayed when a conference is being recorded.	Recording
conference_ locked_ indicator_text	<p>The text that is associated with the conference locked indicator and that is temporarily displayed when a conference is locked.</p> <p>This has a maximum limit of 20 characters.</p>	Conference locked
test_call_ service_ media_delay	<p>The number of seconds that media is delayed before being looped back to the caller when using a Test Call Service.</p> <p>(For more information, see Configuring the Test Call Service.)</p>	2
disable_ streaming_ indicator	Determines whether the streaming indicator icon is disabled (true) or enabled (false).	false
disable_ conference_ locked_ indicator	Determines whether the conference locked and conference unlocked indicators are disabled (true) or enabled (false).	false
streaming_ indicator_text	The text that is associated with the streaming indicator and that is temporarily displayed when a conference is being streamed.	Streaming enabled
public_ streaming_ indicator_text ◇	The text that is associated with the streaming indicator and that is temporarily displayed when Google Meet public streaming is enabled.	Public streaming enabled
enable_dtmf_ conference_ control	Determines whether Host participants can use DTMF to control the conference (true) or not (false).	true
disable_ watermark_ icon	Determines whether the watermark icon is overlaid onto the main speaker video (false) or is not used (true). See Video watermarking for more information.	false
plus_n_ indicator_bg_ color †	The background color of the thumbnail that shows how many additional participants are in the conference. This is used in conjunction with <code>plus_n_indicator_text_color</code> .	0x323232 (dark gray)

Name	Description	Value in Base theme
test_call_service_disconnect_timeout	The number of seconds that a user can test their media before the disconnect message is played, when using a Test Call Service.	10
theme_version	Controls whether legacy (version "1") or new style (version "2") themes are used. If <code>"theme_version": 2</code> is not present in a <code>themeconfig.json</code> file at the lowest level of the theme file hierarchy then it is assumed to be a legacy-style theme (see How do I know which files and configuration settings will be used in a particular VMR? for more information).	2
conference_unlocked_indicator_text	The text that is associated with the conference unlocked indicator and that is temporarily displayed when a conference is unlocked.	Conference unlocked
conference_locked_indicator_n_waiting_text †	The text that is associated with the conference locked indicator text, and shows the number of participants waiting to join the locked conference.	{number_of_waiting_participants} waiting for host
transcribing_indicator_text	The text that is associated with the conference transcribing indicator and that is temporarily displayed when a conference is being transcribed.	Transcribing

† This setting has no effect in the Adaptive Composition layout.

◇ Note that Google Meet public streaming has not been launched by Google yet.

Additional settings

This table contains any additional settings that are not included in the Base `themeconfig.json` file that you download from the Management Node, but can be used to control theme behavior.

Name	Description	Default behavior
enable_solo_streaming_loopback	Normally, the <code>streaming_in_progress</code> splash screen is shown if there are no other participants in the conference other than a streaming participant. If you want to show a loopback of the presentation stream instead of this splash screen you need to insert the setting: <code>"enable_solo_streaming_loopback" : true</code> into your <code>themeconfig.json</code> file.	false
disable_comfort_noise	Low-level, almost imperceptible background noise is added to the audio mix in conferences. This can be disabled by updating your theme configuration — either the default theme, or just in themes applied to specific services. To disable comfort noise you need to insert the setting: <code>"disable_comfort_noise" : true</code> into your <code>themeconfig.json</code> file.	false


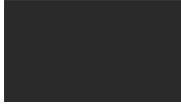
Customizing the splash screens

The splash screens are used when joining a conference (such as to capture a PIN), using a Virtual Reception or the Test Call Service, or to display an error such as insufficient licenses. Each splash screen consists of a background JPG file plus other SVG images and text that are overlaid onto the background image.

The `themeconfig.json` file offers a lot of flexibility when customizing the appearance of the splash screens. You can override the default background image, graphics and text [elements](#) that are used on each splash screen, and modify where those elements are positioned. Each splash screen can be configured by specifying a `splash_screens` object and associated elements within your `themeconfig.json` file, as described below.

Background image

By default the Base theme contains 2 background files:

File name	Image size (width, height)	Content in Base theme	Notes
background.jpg	1920 x 1080 pixels JPG (RGB mode only)		The background image used by default on all splash screens except for those used by the Test Call Service.
background_test_call.jpg	1920 x 1080 pixels JPG (RGB mode only)		The background image (a black screen) used by default on the Test Call Service splash screens.

You can replace either of these files with your own versions, or you have the flexibility to specify an alternative file to use on a specific splash screen. Ensure that the images display acceptably on both 16:9 endpoints and 4:3 endpoints (if applicable). For example, the original 1920 x 1080 pixel (16:9) images may have approximately 240 pixels cropped from both the left and right sides when displayed as 4:3. You should therefore check that any details on the far left or far right of the image are visible in both formats; for this reason we recommend keeping the image fairly central.

To specify an alternative file, use the `background` element to define the override and the `path` element to define the file to use. The filename must begin with "background". For example, to use an alternative background for the `pin_entry` splash screen:

```
{
  "splash_screens": {
    "screens": {
      "pin_entry": {
        "background": {"path": "background_pin_entry.jpg"},
        ...
      }
    }
  }
}
```

Make sure that you include the specified background file (background_pin_entry.jpg in this example) in the theme ZIP file that you upload.

Alternatively, instead of specifying a `path` you can define a color, e.g. `"background": {"color": "0x323232"}`

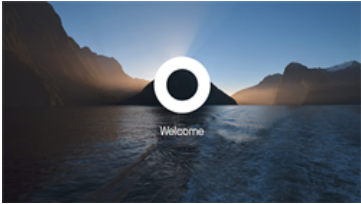

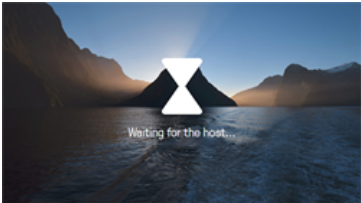


Splash screens

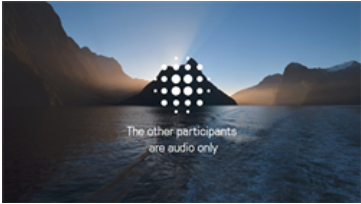
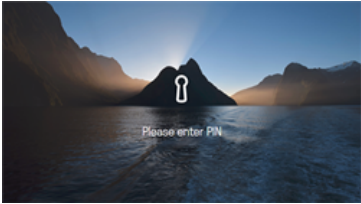

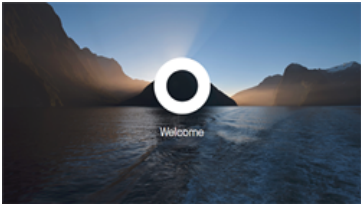


The following table lists the splash screens that you can configure within a theme. It describes when that screen is shown and shows its default appearance. It also identifies the screen key to use in a `screens` object if you want to override the default behavior, and the default `icon image` and `text label` elements used on that screen.







All of the screens used when joining a conference use `background.jpg` as the default background image. The Test Call Service uses `background_test_call.jpg` during the call.

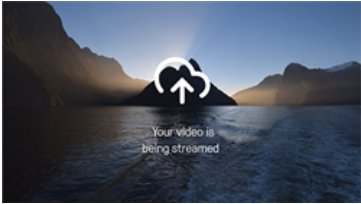
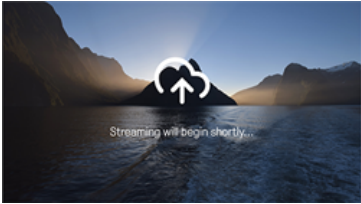




Default splash screens and associated elements


If you are using a theme to customize gateway calls only, then you only need to consider a subset of these screens, as indicated in the table below.

Screen key	Default background, icon, text and layout	Default elements used	Usage	Used in gateway calls
welcome		Icon: icon_welcome.svg Label: welcome	Shown when there are no other participants in the conference. Also shown to a VTC-based participant that is being held in a Skype for Business meeting lobby.	
connecting		Icon: icon_connecting.svg Label: connecting	Shown briefly when placing a person-to-person call via the Infinity Gateway.	✓
waiting_for_host		Icon: icon_waiting_for_host.svg Label: waiting_for_host	Shown to Guests while waiting for a Host to either: <ul style="list-style-type: none"> join the conference allow the Guest to join a locked conference, either by unlocking the conference or permitting the individual participant to join. <p>The length of time Guests can remain at this screen is configurable via the global Waiting for Host timeout option (Platform > Global Settings > Service Configuration).</p>	
inlobby_status_unknown		Icon: icon_connecting.svg Label: inlobby_status_unknown	Shown when initially connecting to a Microsoft Teams meeting, while the participant's status is not yet known.	
inlobby		Icon: icon_is_in_lobby.svg Label: inlobby	Shown while being held in the Microsoft Teams lobby when joining a Microsoft Teams meeting.	

Screen key	Default background, icon, text and layout	Default elements used	Usage	Used in gateway calls
other_participants_audio_only		Icon: icon_other_participants_audio_only.svg Label: other_participants_audio_only	Shown when all other participants are audio-only or presentation and control-only.	✓
pin_welcome		Icon: icon_pin_entry.svg Label: please_enter_pin	The pin_ screens are shown to participants [†] when entering PINs. Participants entering PINs via these screens will be disconnected after three unsuccessful attempts. The length of time participants can remain at these screens is configurable via the global PIN entry timeout option (Platform > Global Settings > Service Configuration).	
pin_entry		Icon: icon_pin_entry.svg and icon_pin_entry_digit.svg	An icon_pin_entry_digit.svg icon is displayed as each number is entered, which can go up to 20 icons over 2 lines. See PIN digits element for controlling the layout of the PIN entry digit icons.	
pin_correct		Icon: icon_welcome.svg Label: pin_correct	Displayed when a correct PIN is entered.	
pin_invalid		Icon: icon_pin_invalid.svg Label: invalid_pin	Displayed when an incorrect PIN is entered.	
virtual_reception_welcome		Icon: icon_virtual_reception_welcome.svg Label: enter_conference_id	Shown to participants [†] who have connected to a Virtual Reception.	

Screen key	Default background, icon, text and layout	Default elements used	Usage	Used in gateway calls
virtual_reception_conference_id_entry		Icon: icon_virtual_reception_conference_id_entry.svg Label: conference_id_entry	Shown to participants† as they enter digits ("1234" shown in this example) into a Virtual Reception.	
virtual_reception_conference_id_invalid		Icon: icon_virtual_reception_conference_id_invalid.svg Label: conference_id_invalid	Shown to participants† who have connected to a Virtual Reception and have entered an invalid conference number.	
virtual_reception_connecting		Icon: icon_virtual_reception_connecting.svg Label: virtual_reception_connecting	Shown briefly to participants† while being transferred from a Virtual Reception. Note that the name of the destination VMR or matching Call Routing Rule is also overlaid onto this image, using the virtual_reception_connecting text label.	
timeout		Icon: icon_timeout.svg Label: timeout	Shown briefly prior to disconnecting participants† who have connected to a Virtual Reception but did not enter a valid conference number.	
no_incoming_video		Icon: icon_no_video.svg	Shown when a participant's video stream is not available.	✓
no_main_video		Icon: icon_no_video.svg	Used when streaming a conference to indicate that there is currently no main video. The no_incoming_video and no_main_video screens both use the same icon image file (icon_no_video.svg) and the default background color is #323232 (dark gray), instead of background.jpg.	

Screen key	Default background, icon, text and layout	Default elements used	Usage	Used in gateway calls
streaming_in_progress	 A dark blue background with a mountain range. In the center, there is a white icon of two people with arrows pointing towards each other. Below the icon, the text "Your video is being streamed" is displayed.	Icon: icon_streaming_screen.svg Label: streaming_in_progress	Shown if there are no other participants in the conference other than a streaming participant. Alternatively, if you want to show a loopback of the presentation stream instead of this splash screen you need to insert <code>"enable_solo_streaming_loopback" : true</code> into your <code>themeconfig.json</code> file (see Additional settings).	
stream_waiting	 A dark blue background with a mountain range. In the center, there is a white icon of two people with arrows pointing towards each other. Below the icon, the text "Streaming will begin shortly..." is displayed.	Icon: icon_streaming_screen.svg Label: stream_waiting	A "holding" splash screen that can be sent to a streaming participant (e.g. YouTube broadcast) while you are waiting for people in the conference to get ready to start.	
no_presentation	 A dark blue background with a mountain range. In the center, there is a white icon of a presentation screen. Below the icon, the text "Waiting for presentation..." is displayed.	Icon: icon_no_presentation.svg Label: no_presentation	Used to indicate that there is currently no presentation stream.	
error_capacity_exceeded	 A dark blue background with a mountain range. In the center, there is a white icon of a triangle. Below the icon, the text "Capacity exceeded" is displayed.	Icon: icon_error.svg Label: capacity_exceeded	Shown to participants† when they attempt to join a conference that has reached its maximum number of participants. For more information, see Limiting the number of participants .	
error_insufficient_licenses	 A dark blue background with a mountain range. In the center, there is a white icon of a triangle. Below the icon, the text "Insufficient licenses" is displayed.	Icon: icon_error.svg Label: insufficient_licenses	Shown to participants† when they are unable to join a conference because all call licenses are currently in use. For more information, see Insufficient licenses .	
error_insufficient_video_licenses	 A dark blue background with a mountain range. In the center, there is a white icon of a triangle with a diagonal line through it. Below the icon, the text "Insufficient video licenses. You can still share content and audio" is displayed.	Icon: icon_error_insufficient_video_licenses.svg Label: insufficient_video_licenses	Shown to participants when they are unable to join a conference because all port (video) licenses are currently in use, but there was an audio license available. For more information, see Insufficient licenses .	✓

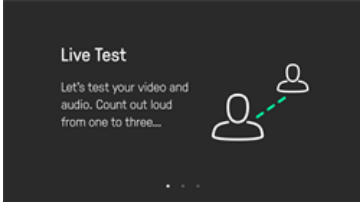
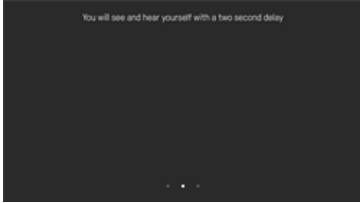
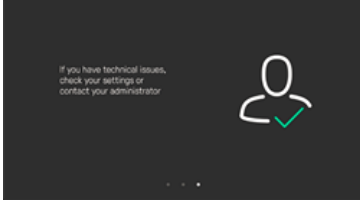
Screen key	Default background, icon, text and layout	Default elements used	Usage	Used in gateway calls
error_invalid_license	 A splash screen with a dark blue background showing a white mountain peak and the text "Invalid license" in the center.	Icon: icon_error.svg Label: invalid_license	Shown to participants [†] when they are unable to join a conference because the Pexip Infinity license is not valid. For more information, see Invalid license .	

[†] Except for participants that are using Infinity Connect clients.

^{††} Only applies to gateway calls into an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.

[‡] Only applies to gateway calls into Microsoft Teams meetings.

These are the splash screens used by the Test Call Service:

Screen key	Default background, icon, text and layout	Default elements used	Usage
test_call_welcome	 A dark blue splash screen with the text "Live Test" and "Let's test your video and audio. Count out loud from one to three..." followed by a small icon of two people connected by a dashed line.	Icon: icon_test_call_welcome.svg Label: test_call_welcome_header and test_call_welcome_text	Shown at the start of a call to a Test Call Service. (For more information, see Configuring the Test Call Service .)
test_call_in_progress	 A dark blue splash screen with the text "You will see and hear yourself with a two second delay" and a small icon of a person with a checkmark.	Label: test_call_in_progress	Shown during a call to a Test Call Service. Note that a large, live (with a short delay) video image of the test call participant is shown on top of this screen during a test call.
test_call_complete	 A dark blue splash screen with the text "If you have technical issues, check your settings or contact your administrator" and a small icon of a person with a checkmark.	Icon: icon_test_call_complete.svg Label: test_call_complete	Shown briefly prior to automatically disconnecting the participant from a Test Call Service.

Text overlays (labels)

The `labels` object is referenced by every text element that is overlaid onto a splash screen and is used to define the actual text displayed to users. For example:

```
"splash_screens": {
  "labels": {
    "welcome": "Welcome",
    "connecting": "Connecting",
    "timeout": "Time exceeded",
    ...
  }
}
```

The keys (e.g. "timeout") are used in [text elements](#) and the associated values (e.g. "Time exceeded") are the text strings that are shown on the screen. The text content can include \n newlines if you require multi-line capability (newlines must be explicitly used and are never applied automatically). On some splash screens, strings may also reference some [predefined variables](#), such as "{conference_name}" which is used in the virtual_reception_connecting screen.

The default font size is 66px and the default color is 0xffffffff (white, defined in ARGB hexadecimal notation).

Note that the default text [labels](#) are not contained within the Base theme's `themeconfig.json` file that you can download from the Management Node, but you can override the default settings by adding the relevant labels into your own `themeconfig.json` file. The following table contains the default keys, the default text string for each key, and the screen in which that key is used:

Default label keys and usage

These are the label keys and associated text strings and screens:

Label key	Default text string (to overlay onto the screen background)	Associated screen (by default)
welcome	Welcome	welcome
connecting	Connecting	connecting
timeout	Time exceeded	timeout
waiting_for_host	Waiting for the host...	waiting_for_host
other_participants_audio_only	The other participants\n are audio only	other_participants_audio_only
please_enter_pin	Please enter PIN	pin_welcome
pin_correct	Welcome	pin_correct
invalid_pin	Invalid PIN\nPlease try again	pin_invalid
enter_conference_id	Enter conference ID	virtual_reception_welcome
conference_id_entry	{conference_id}	virtual_reception_conference_id_entry
conference_id_invalid	Conference ID invalid	virtual_reception_conference_id_invalid
virtual_reception_connecting	{conference_name}	virtual_reception_connecting
no_presentation	Waiting for presentation...	no_presentation
streaming_in_progress	Your video is\nbeing streamed	streaming_in_progress
stream_waiting	Streaming will begin shortly...	stream_waiting
inlobby_status_unknown	Call connecting	inlobby_status_unknown
inlobby	Welcome to the lobby,\nyour meeting host will admit you soon	inlobby
capacity_exceeded	Capacity\nexceeded	error_capacity_exceeded
insufficient_licenses	Insufficient licenses	error_insufficient_licenses
insufficient_video_licenses	Insufficient video licenses\nYou can still share content and audio	error_insufficient_video_licenses
invalid_license	Invalid license	error_invalid_license
test_call_welcome_header	Live Test	test_call_welcome

Label key	Default text string (to overlay onto the screen background)	Associated screen (by default)
test_call_welcome_text	Let's test your video and\audio. Count out loud\nfrom one to three...	test_call_welcome
test_call_in_progress	You will see and hear yourself with a two second delay	test_call_in_progress
test_call_complete	If you have technical issues,\ncheck your settings or\ncontact your administrator	test_call_complete

Adding your own text strings and translations

Adding your own text strings

You can specify your own keys and text strings within the `labels` object and use them on any splash screen. For example, to specify a new label and use the associated text and an alternative graphic on the `welcome` screen:

```
"splash_screens": {
  "labels": {
    "my_label_key": "¡Hola!"
  },
  "screens": {
    "welcome": {
      "elements": [
        {"type": "icon", "path": "icon_my_welcome_image.svg"},
        {"type": "text", "label": "my_label_key"}
      ]
    }
  }
  ...
}
```

Translations

If you want to use a theme where all of the strings are translated to a different language, you have the ability to change the wording of the text without modifying the layout itself, for example:

```
"splash_screens": {
  "labels": {
    "welcome": "Velkommen",
    "connecting": "Kobler til",
    ...
  }
}
```

Predefined variables to use in text strings

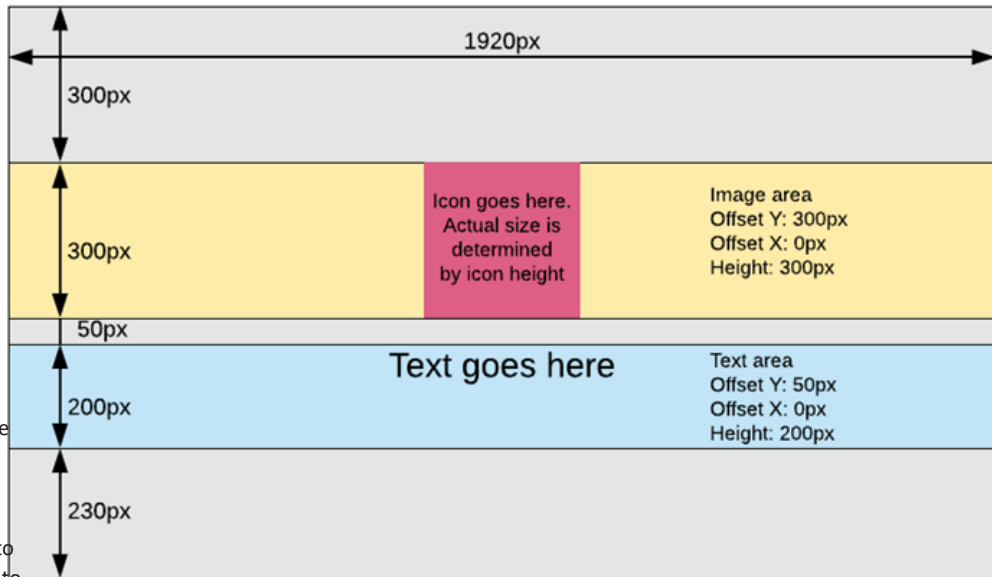
There is a small set of predefined variables that you can reference within your text strings:

Variable name	Description and usage
{conference_id}	This variable is only supported on the <code>virtual_reception_conference_id_entry</code> screen, where it is normally used by the <code>conference_id_entry</code> label and it displays the number the user is entering into the Virtual Reception.
{conference_name}	Contains the name of the conference. This variable is used on the <code>virtual_reception_connecting</code> screen, where it is included in the <code>virtual_reception_connecting</code> label and it refers to the conference into which the user is being transferred. For example: <pre>"labels": {"virtual_reception_connecting": "{conference_name}"}</pre> This variable can also be used in labels on the <code>welcome</code> and <code>waiting_for_host</code> screens.
{number_of_waiting_participants}	Used while a conference is in progress and contains the number of participants waiting to be let into a locked conference. It is normally used with the <code>conference_locked_indicator_n_waiting_text</code> setting in the <code>themeconfig.json</code> file (note that this is a setting and not a screen label). For example: <pre>"conference_locked_indicator_n_waiting_text": "{number_of_waiting_participants} waiting for host"</pre>

Splash screen elements (to control the size and position of text/graphics)

When customizing a splash screen you can define `elements` to control the appearance and position of images and text on that specific screen. The default screen layout and offsets are shown in this diagram (right).

i If you define `elements` for a splash screen, whatever you specify will completely replace all of the default icon and text elements for that screen. For example, if you want to add a new line of text to a screen, your `elements` object for that screen must define the original (default) icon and text as well as your new text.



The following example shows the JSON structure for how you can define an `icon` and `text` type element for a splash screen (the `welcome` screen in this example). All width, height and offset positions are measured in pixels.

```
{
  "theme_version": 2,
  "splash_screens": {
    "screens": {
      "welcome": {
        "elements": [
          {
            "type": "icon",
            "path": "icon_welcome.svg",
            "offset_y": 300,
            ...
          },
          {
            "type": "text",
            "label": "welcome",
            "font_size": 66,
            ...
          }
        ]
      },
      ...
    }
  },
  ...
}
```

The full set of keys that you can specify for each element type are described below.

Icon elements (for SVG files used on splash screens)

Icons are the SVG files that are overlaid onto a splash screen. They can be configured by the `icon` element, which controls the position and scale (size) of the SVG file, and which file to use.

To change the color of an SVG image, see [Changing the color of the SVG image](#).

Example icon element usage

This shows how to use the `icon` element and its associated keys.

```
{
  "type": "icon",
  "path": "icon_virtual_reception_welcome.svg",
  "width": 1920,
```



```

"height": 300,
"offset_y": 300,
"offset_x": 0
}

```

Key	Description
path	<p>Name of the SVG file (where the filename matches the pattern "icon_[A-Za-z0-9_].svg").</p> <p>This can be the name of a Base SVG file (where you can also optionally include a replacement file of the same name in the theme ZIP), or a new filename (where you must supply the associated SVG file of that name).</p>
width and height	<p>Icon images are rendered proportionally scaled using the smallest value of width or height.</p> <p>The icon width defaults to 1920, which means that the icon size is normally controlled by the height. Width is normally specified only in a free-form layout when you want to restrict icon size based on width to ensure that there is no overflow on the x axis.</p> <p>The icon height is usually the attribute that controls the size of the icon. It defaults to 300px for most screens. If height is set to a larger value than width, then width controls the size of the icon. In vertical layouts, height is also used to calculate the y-axis coordinate of the element below it.</p> <p>Ultimately, the proportions of the image itself control how it is rendered, within a certain width or height constraint. Note that the icon is centered inside the bounding box; for example if you specify <code>{"width": 1920, "height": 200}</code> for a 1000x200 icon, it will display centered as expected, but if you then add, for example, <code>offset_x: 500</code> the icon will get pushed to the right and cropped.</p>
offset_x	The x-axis coordinate relative to the center in vertical layouts, and absolute in <code>free_form</code> layouts. It can be negative.
offset_y	The y-axis coordinate relative to the element above (or top edge) in vertical layouts, and absolute (from the top of the screen) in <code>free_form</code> layouts. It can be negative.

Text elements

You have full control over the positioning and style of a text string by using a `text` element. All measurements are in pixels. The default font for the in-conference display of participant names is Roboto (which cannot be changed), or if that is not available for the character set, Noto Sans. All other overlay text (splash screen text and in-conference messages) uses GT Pressura Light and this can be changed by using a customized theme. For example (and showing the default values for non-test-call conferences):

Text element usage

This shows how to use the `text` element and its associated keys.

```

{
  "type": "text",
  "label": "enter_conference_id",
  "offset_y": 50,
  "offset_x": 0,
  "height": 200,
  "width": 1920,
  "color": "0xffffffff",
  "font_size": 66,
  "font": "GT Pressura Light",
  "letter_spacing": 0,
  "line_spacing": -13,
  "align": "center"
}

```

Key	Description
label	A reference to an item in the labels object , for example "enter_conference_id", to identify the actual text to display.
offset_x	The x-axis coordinate offset of the bounding box in vertical layouts, and the absolute coordinate of the bounding box in <code>free_form</code> layouts.

Key	Description
offset_y	The y-axis coordinate offset (from the top of the screen) of the bounding box in vertical layouts, and the absolute coordinate of the bounding box in free_form layouts.
width	Width of the bounding box. It usually defaults to 1920 in vertical layouts.
height	Height of the bounding box.
color	<p>Color of the text. You can use either:</p> <ul style="list-style-type: none"> RGB hexadecimal notation in the format 0xnnnnnn ARGB hexadecimal notation (0xnnnnnnnn), where the first two digits after the 0x represent the Alpha channel, indicating the level of opacity applied. Example opacity values include: 00 - fully transparent, 80 - 50% opaque, BE - 75% opaque, FF - fully opaque. <p>Default: 0xffffffff (white, fully opaque).</p> <p>You can specify a default color for all text elements by adding a <code>text_color</code> setting to the <code>themeconfig.json</code> file, for example adding <code>"text_color": "0xff0000"</code> would make all text elements red by default.</p>
font_size	The font size in pixels. Default: 66px.
font	The font name. The supported fonts are: "Noto Sans", "Roboto", "Roboto Condensed", "GT Pressura", "GT Pressura Light".
letter_spacing	The spacing between letters. It can be negative. Default: 0.
line_spacing	The spacing between lines. It can be negative. Default: -13.
align	Alignment of text within the bounding box. Possible values: "center", "left", "right". Default: "center".

PIN digits element

The `pin_digits` element is a special element to render up to 20 icons (each 42x42 pixels by default) on 2 lines, and is only used for PIN entry on the `pin_entry` screen. For example (showing the default values):

PIN digits element usage

This shows how to use the `pin_digits` element and its associated keys.

```
{
  "type": "pin_digits",
  "path": "icon_pin_entry_digit.svg",
  "digit_width": 42,
  "digit_height": 42,
  "digit_spacing": 10,
  "offset_y": 50,
  "offset_x": 0,
  "height": 200,
  "width": 1920,
  "line_spacing": 25
}
```

Key	Description
path	Name of the PIN digit icon SVG file.
digit_width	The width of each individual PIN digit icon.
digit_height	The height of each individual PIN digit icon.

Key	Description
digit_spacing	The space between each PIN digit icon.
offset_x	The x-axis offset of the bounding box.
offset_y	The y-axis offset (from the top of the screen) of the bounding box.
height	Height of the bounding box.
width	Width of the bounding box.
line_spacing	Spacing between the lines of digits. There are always (up to) 10 digits per line, regardless of the size of the bounding box. You must ensure that the size of the bounding box is big enough to fit the digit icons (based on the maximum length of the PINs used in your deployment).

Vertical and free form layouts

The splash screens support 2 types of layout: vertical (the default) and free form. Layouts are specified by the `layout_type` entry in the screen specification object.

Using vertical and free form layouts

For example, to specify a vertical layout for the `error_invalid_license` screen:

```
{
  "splash_screens": {
    "screens": {
      "error_invalid_license": {
        "layout_type": "vertical"
      }
    }
  }
}
```

Vertical layouts

Vertical layouts are centered layouts where the elements (icons and text) stack on top of each other top to bottom in the order they have been specified. The `height` and `offset_y` attributes determine the element's placement on the screen. All elements are centered by default and can be shifted left and right with the `offset_x` variable. For example:

```
"splash_screens": {
  "screens": {
    "welcome": {
      "layout_type": "vertical",
      "elements": [
        { "type": "icon", "path": "icon_welcome.svg", "height": 300, "offset_y": 400 },
        { "type": "text", "text": "welcome", "height": 200, "offset_y": 50 }
      ]
    }
  }
}
```

In this layout, the icon is centered on the screen, starting 400px from the top of the screen. The text element is placed 50px under the icon, placing it at $y=750\text{px}$ ($300\text{px height} + 400\text{px offset of the icon} + 50\text{px offset of the text}$).

Free form layouts

Free form layouts are, as the name implies, completely free form. This lets you design complex layouts, like the one used for the Test Call Service. In a free form layout, `offset_x` and `offset_y` are absolute coordinates on the screen and you are responsible for placing the elements correctly.

This example shows the layout of the `test_call_complete` screen (these are the actual values that are used by default):

```
{
  "theme_version": 2,
  "splash_screens": {
    "screens": {
      "test_call_complete": {
        "layout_type": "free_form",
        "background": { "path": "background_test_call.jpg" },
        "elements": [
          {
            "type": "text",
            "label": "test_call_complete",

```

```
        "height": 300,
        "offset_y": 361,
        "offset_x": 308,
        "align": "left",
        "font_size": 50
    },
    {
        "type": "icon",
        "path": "icon_test_call_complete.svg",
        "offset_x": 1259,
        "offset_y": 305,
        "height": 386,
        "width": 380
    },
    {
        "type": "icon",
        "path": "icon_inactive_page.svg",
        "offset_x": 872,
        "offset_y": 985,
        "width": 16,
        "height": 16
    },
    {
        "type": "icon",
        "path": "icon_inactive_page.svg",
        "offset_x": 952,
        "offset_y": 985,
        "width": 16,
        "height": 16
    },
    {
        "type": "icon",
        "path": "icon_active_page.svg",
        "offset_x": 1032,
        "offset_y": 985,
        "width": 16,
        "height": 16
    }
  ]
}
}
```

Helper lines

To aid development, debugging and theme creation, you can display gridlines for a given splash screen by using the `helper_lines` object. Helper lines are added on top of the background and under the icons, text and other elements.

How to add help lines

Example use of helper lines on the `error_invalid_license` screen:

```
{
  "splash_screens": {
    "screens": {
      "error_invalid_license": {
        "helper_lines": {
          "horizontal": [300, 350, 432, 678],
          "vertical": [660, 960, 1260]
        }
      }
    }
  }
}
```

Key	Description
horizontal	A list of coordinates (from the top of the screen) to put horizontal helper lines.
vertical	A list of coordinates to put vertical helper lines.







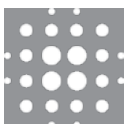


Overlay SVG image files used on splash screens when joining a conference









When creating your own SVG image files for use on the splash screens, do **not** use tools such as Adobe Photoshop or Illustrator to convert JPG or PNG files into SVG format as they typically create complex files/paths that are resource-intensive to render, or in some cases may render incorrectly.

- Use a vector drawing program that uses SVG as its primary format, such as Inkscape.
- Do not create a single complex path — use multiple paths instead.

Default SVG image files



All of the overlay vector images in the Base theme are listed below. All of the images are white, except for `icon_inactive_page.svg`, and are shown here against a gray background. The table also lists the [splash screens](#) on which the overlay images are used by default.

File name	Content in Base theme	Associated default splash screens
<code>icon_active_page.svg</code>		test_call_welcome test_call_in_progress test_call_complete (they are rendered as the progress dots along the bottom of the screen)
<code>icon_connecting.svg</code>		connecting inlobby_status_unknown
<code>icon_error.svg</code>		error_capacity_exceeded error_insufficient_licenses error_invalid_license
<code>icon_error_insufficient_video_licenses.svg</code>		error_insufficient_video_licenses
<code>icon_inactive_page.svg</code>		test_call_welcome test_call_in_progress test_call_complete (they are rendered as the progress dots along the bottom of the screen)
<code>icon_is_in_lobby.svg</code>		inlobby
<code>icon_other_participants_audio_only.svg</code>		other_participants_audio_only
<code>icon_pin_entry.svg</code>		pin_welcome pin_entry
<code>icon_pin_entry_digit.svg</code>		pin_entry

File name	Content in Base theme	Associated default splash screens
icon_pin_invalid.svg		pin_invalid
icon_streaming_screen.svg		streaming_in_progress stream_waiting
icon_timeout.svg		timeout
icon_virtual_reception_conference_id_entry.svg		virtual_reception_conference_id_entry
icon_virtual_reception_conference_id_invalid.svg		virtual_reception_conference_id_invalid
icon_virtual_reception_connecting.svg		virtual_reception_connecting
icon_virtual_reception_welcome.svg		virtual_reception_welcome
icon_waiting_for_host.svg		waiting_for_host
icon_welcome.svg		welcome pin_correct

Overlay SVG image files used by the Test Call Service

These overlay SVG image files are used only by the Test Call Service:

File name	Content in Base theme	Associated default splash screens
icon_test_call_complete.svg		test_call_complete
icon_test_call_welcome.svg		test_call_welcome

Changing the color of the SVG image

The color in an SVG file is typically controlled by a `fill` property defined within the file itself. Most of the SVG images used in the Base Pexip theme are white (`fill="#FFFFFF"`).

To change the color of an SVG image you can edit the SVG file in a simple text editor such as Notepad, and change the fill color as required. For example to change the color to red, you would specify `fill="#f00000"`.




Indicator graphics used within an ongoing conference









The following graphics may be used during a conference. The SVG indicators are rendered on top of a dark gray background (the color is 0x7f323232, where 7f is the alpha channel which makes the background slightly transparent). The gray background is not part of the actual SVG image itself and it cannot be customized. If you use your own customized SVG images, ensure that they look appropriate when displayed on sizes from 20x20px to 48x48px, and are recognizable when displayed as 12x12px for participants that use very low resolutions.

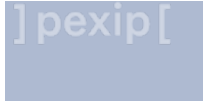
If you are using a theme to customize gateway calls only, then you only need to consider a subset of these images, as indicated in the table below.

Default in-conference graphics files

These are the default graphics files that used during a conference:

File name	Content in Base theme	Notes	Used in gateway calls
icon_audio.svg		Used to represent an audio-only participant. For more information, see Features common to all layouts .	✓
icon_conference_locked.svg		Slides out over the main video image (displayed to Hosts only) to indicate that the conference is currently locked . It is used in conjunction with the <code>conference_locked_indicator_text</code> and the <code>conference_locked_indicator_n_waiting_text</code> in the <code>themeconfig.json</code> file.	
icon_conference_unlocked.svg		Slides out over the main video image (displayed to Hosts only) to indicate that the conference has been unlocked . It is used in conjunction with the <code>conference_unlocked_indicator_text</code> in the <code>themeconfig.json</code> file.	

File name	Content in Base theme	Notes	Used in gateway calls
icon_no_presentation.svg		Used to indicate that there is currently no presentation stream. It is used by default on the no_presentation splash screen.	
icon_no_video.svg		Shown when a video stream is not available. It is used by default on the no_main_video and no_incoming_video splash screens. It should look appropriate when displayed at sizes up to 400 x 400 pixels.	✓
icon_video_muted.svg		Used to represent a video-muted participant who is excluded from the video layout.	✓
icon_notch57tp_audio.svg icon_notch57tp_conference_locked.svg icon_notch57tp_participants.svg icon_notch57tp_public_streaming.svg icon_notch57tp_streaming.svg icon_notch57tp_video_muted.svg	<various>	These icon images are used to support the Adaptive Composition layout, instead of the other in-call indicators shown here. We recommend that you do not customize these elements as we cannot guarantee compatibility with future versions of Pexip Infinity.	
presence_avatar_image.jpg		Shown next to the conference alias when using legacy Infinity Connect clients, and as the contact avatar when using Skype for Business / Lync. When the image is used, the corners are cropped so that it can be displayed as a round image. It is a JPG image (RGB mode only) and should be 128 x 128 pixels.	✓
icon_recording.svg		Indicates that the conference is being <u>recorded</u> .	
icon_streaming.svg		Indicates that the conference is being <u>streamed</u> .	
icon_public_streaming.svg ◊		Indicates that the conference is being publicly streamed via a Google Meet conference.	
icon_transcribing.svg		Indicates that a Microsoft Teams conference is being transcribed.	

File name	Content in Base theme	Notes	Used in gateway calls
watermark_icon.png		<p>Transparent image used for applying a watermark to the main speaker video in a conference.</p> <p>The default image is a white Pexip logo with 40% transparency and is shown here against a blue background.</p> <p>It is a PNG image and should be 200 pixels wide x 100 pixels high.</p> <p>See Video watermarking for more information.</p>	

† Only applies when the gateway call is into an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet. In Google Meet conferences, the `icon_recording.svg` image is used when the conference is being recorded and the `icon_streaming.svg` image is used when the conference is being streamed. The `icon_public_streaming.svg` icon is used only in Google Meet conferences.

‡ Only applies to gateway calls into Microsoft Teams meetings.

◇ Note that Google Meet public streaming has not been launched by Google yet.

Video watermarking

Video watermarking overlays a small transparent image onto the main speaker video during a conference (VMRs and Virtual Auditoriums). Watermarks are never added to calls placed via the Infinity Gateway.

The default watermark is a white "Pexip" logo and is enabled by default. You can change the watermark image or disable watermarking completely.

Enabling, disabling and changing the watermark image

When watermarking is enabled:

- The `watermark_icon.png` file is used as the watermark image.
- The top-left corner of the watermark is placed 56 pixels across and 40 pixels down from the top-left corner of the main speaker video, and the watermark area then extends 200 pixels across and 100 pixels down. These sizes are based on Full HD resolution — they will be smaller on lower resolutions.
- The size and position of the watermark area cannot be customized.
- The watermark is shown on the following splash screens: `welcome`, `connecting`, `streaming_in_progress`, `stream_waiting`, `other_participants_audio_only`, `no_presentation`, `no_incoming_video`, `no_main_video`.

Enabling or disabling watermarking

The `disable_watermark_icon` setting in the `themeconfig.json` file controls whether watermarking is enabled or not:

- Set `disable_watermark_icon` to *false* to enable watermarking.
- Set `disable_watermark_icon` to *true* to disable watermarking.

By default, `disable_watermark_icon` is set to *false*.

Changing the watermark image

When creating your own watermark file (`watermark_icon.png`) for inclusion in a theme:

- The file must be PNG format and include a transparent image (if it is not transparent the overlaid content will be fully opaque).
- We recommend that the watermark image is white with 40% transparency.
- The image should be 200 pixels wide x 100 pixels high. Larger images are automatically scaled to fit to the top left corner of the watermark area (and thus may lose some definition).

Themes used by Call Routing Rules (gateway calls)

You can assign a theme to a Call Routing Rule. Only a subset of the theme's image files are used when they are applied to an Infinity Gateway call. Thus, you only need to customize a subset of the files for any themes that are only assigned to one or more Call Routing Rules. The various tables on this page indicate, where appropriate, which image files are used in gateway calls.

Also note that:

- the **audio** files are **not** used in themes assigned to Call Routing Rules.
- from the `themeconfig.json` file, only the `disable_streaming_indicator` and `streaming_indicator_text` / `recording_indicator_text` settings are used (for gateway calls into an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet).

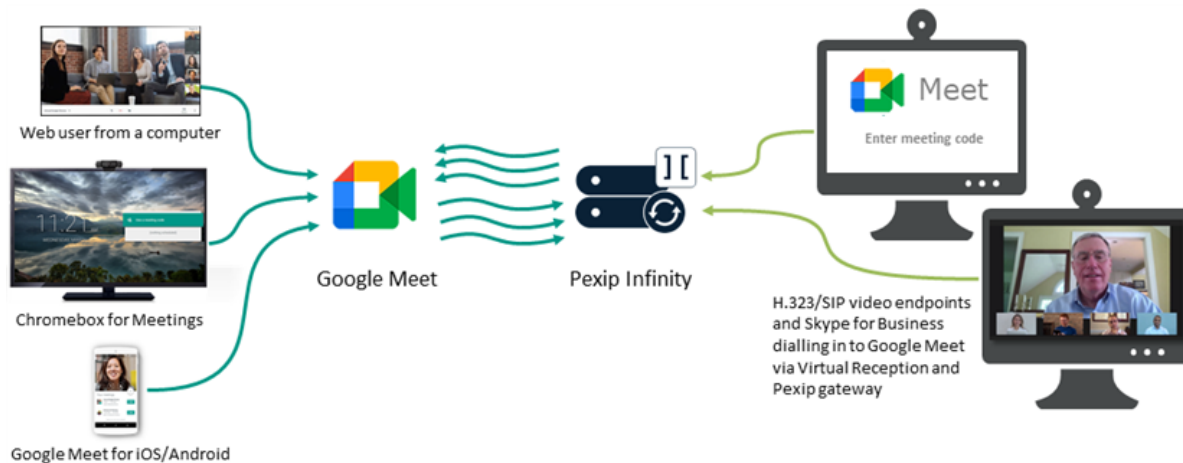
Integrating Google Meet with Pexip Infinity

Introduction

The Infinity Gateway provides any-to-any video interoperability with Google Meet. It enables the seamless interoperation of HD video between Google Meet conferences and:

- H.323 & SIP room-based videoconferencing systems, including Cisco, Poly, LifeSize, and others
- Microsoft Skype for Business and Lync
- Browser-based video (WebRTC / RTMP).

Third-party systems can connect to Google Meet conferences via the Infinity Gateway either by dialing the conference directly or via a Virtual Reception (IVR).



The Google Meet in-call features that are supported via the Infinity Gateway include:

- Active speaker switching
- Content sharing
- Recording indicator
- Bandwidth optimizations

Pexip interoperability can be used with all paid Google Workspace licenses.

Note that Google Meet is inherently a dial-in service i.e. you can only dial **from** a third-party video system into Google Meet. You cannot dial **out** from Google Meet to a SIP, H.323 device etc — instead, you have to send the relevant joining instructions/invitation to the user of that device.

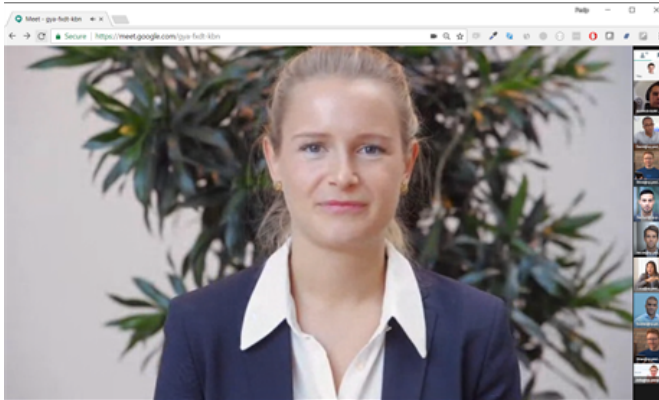
Deployment environments

The Pexip Infinity platform can be deployed in any of its supported environments such as on-premises or in a public or hybrid cloud (including Google Cloud Platform).

Pexip Infinity has a close integration with Google Meet and uses Google Meet APIs to provide Infinity's interoperability features. Even though Pexip strives to maintain backwards compatibility between older versions of Pexip Infinity and the latest release of Google Meet, to ensure compatibility with the latest updates to Google Meet we recommend that you aim to keep your Pexip Infinity deployment up-to-date with the latest Pexip Infinity software release. If, for example, you have a large Pexip deployment for non-Google Meet related services, and you have stringent upgrade procedures meaning that you do not always keep your Infinity software up-to-date with the latest release, you may want to consider deploying a second instance of the Pexip Infinity platform that is dedicated to your Google Meet interoperability requirements, and which can be managed separately and upgraded more frequently.

Native user experience for all participants

All participants receive the appropriate native user experience of the Google Meet conference when there is a mix of direct and gatewayed third-party participants.



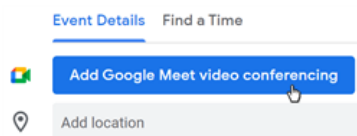
Google Meet experience when third-party VTC systems are connected to the conference



Third-party VTC system experience (Pexip's standard 1+7 layout) when connected to a Google Meet conference

Scheduling and joining conferences

Users in your organization can schedule meetings as normal via their Google Calendar, and choose to add Google Meet conferencing within their event options. The joining instructions for internal and external standards-based video conferencing systems, and Skype for Business users, are then automatically included in, or linked from, the Calendar invitation and event.



All calls are routed into the Google Meet conference by means of the meeting ID that is associated with that conference.

Meeting IDs are generated automatically by Google Meet. The meeting could have a long ID, a short ID, or both a long and short ID, depending on how you

have configured your Google Meet interop settings within the Google Admin console. Long IDs are added to calendar events and invites when events are created; short IDs and Skype for Business joining instructions are available from a "More joining options" link in the calendar event or invite.

For ad hoc conferences, links to the meeting IDs are presented to the host when the meeting is initiated and are also available from the **Meeting details** option while in the conference.

Team meeting

Tuesday, 23 February · 17:00 – 18:00

Join with Google Meet

meet.google.com/bsr-mzid-hsd

Join by phone

(GB) +44 20 3957 0773 PIN: 487 426 342#

5216525442596@example.com

ID: 5216525442596

More joining options

30 minutes before

Enabling access and admitting external participants into Google Meet conferences

After you have installed and performed the basic configuration of your Pexip Infinity platform, you have to link your Pexip platform to your Google Workspace account, so that it can route calls into your Google Meet conferences. This is handled via **access tokens**, which are private codes that can be used by a third-party system, such as Pexip Infinity, to identify your account.

You can set up two types of access tokens in your Google Workspace account: a **trusted** and an **untrusted** token. You can use these two token types to control whether an endpoint that is routed via Pexip Infinity into a Google Meet conference is automatically admitted into the conference, or whether an existing conference participant has to explicitly admit it into the conference. When you configure Pexip Infinity, you decide which type of token to associate with the access rules and dial patterns that allow devices to be routed into Google Meet conferences.

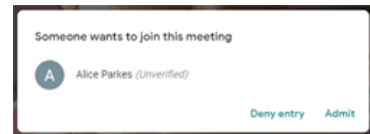
Pexip Infinity also adds an additional layer of trust control by including an explicit setting on each Call Routing Rule to indicate whether or not the devices that are routed via that rule into Google Meet are trusted endpoints from Pexip Infinity's perspective (for example, you could treat the device as trusted if the caller is coming from a specific location, or if the device is registered to Pexip Infinity).

In essence, when Pexip Infinity routes a call to Google Meet, it provides three pieces of information:

- the meeting ID (so that the endpoint joins the correct conference)
- the access token, which can be either a "trusted" or "untrusted" token
- a "domain member" flag, which indicates if the calling endpoint is a trusted endpoint from Pexip Infinity's perspective.

If the access token is a trusted token **and** the endpoint is trusted by Pexip Infinity, then the device is automatically admitted into the conference.

In all other cases, the device has to be explicitly admitted into the conference (this takes the form of a popup dialog as shown right, which is displayed to all participants who are connected directly to the conference). Any of those participants can then choose to allow (admit) or deny access.



See [Configuring Google Workspace for Google Meet integration](#) and [Configuring Pexip Infinity as a Google Meet gateway](#) for details about configuring access tokens, and [Registering devices to Pexip Infinity](#) for registration information.

Presentation / content sharing

If a participant who is using the native Google Meet web client starts presenting, any VTC participants are able to see both the presentation content and the presenter's standard video stream. Similarly, if a VTC participant starts presenting, the other participants in the conference will see both the presentation stream and the video from that participant. Note that in version 21 and earlier of Pexip Infinity, if a VTC participant starts presenting, the other participants in the conference only see the presentation stream from that participant.

Recording and streaming

If a Google Meet conference is recorded or streamed, audio prompts indicating that streaming or recording has been started/stopped are played to callers who are gatewayed via Pexip Infinity into the conference, and distinct messages and indicators are used depending on whether the conference is being recorded, streamed or both. When streaming, the audio prompts and indicators also vary according to whether the stream is public or not.



Configuring Google Workspace for Google Meet integration

This topic explains the Google Workspace configuration steps that are required when integrating Pexip Infinity with Google Meet.

It covers the following processes:

- [Generating your gateway access tokens](#)
- [Google Meet interoperability settings](#)

You can configure your Google Meet settings from the Google Admin console via **Apps > Google Workspace > Google Meet**.

For more information about setting up Google interoperability, see <https://support.google.com/a/answer/7673980>.

Generating your gateway access tokens

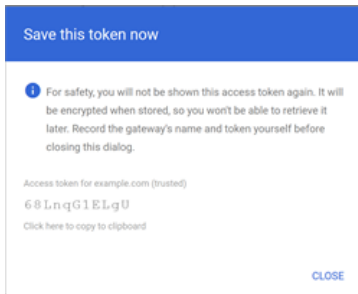
Gateway access tokens are the private codes assigned to your Workspace account that are used by Pexip Infinity when it routes calls into your Google Meet conferences. Tokens can be defined as "trusted" or "untrusted", see [Enabling access and admitting external participants into Google Meet conferences](#) for details.

- i** The generated tokens are only displayed once in Google Workspace, at the time of creating them. These tokens need to be configured on your Pexip Infinity system. Therefore you should have your Pexip Infinity Administrator interface open and available at the same time as you are generating the tokens within Google Workspace.

To set up your trusted and untrusted gateway access tokens from the Google Workspace administrator console:

1. Go to **Apps > Google Workspace > Google Meet > Gateways for Interoperability**.
2. Select **Add Gateway**.
3. Enter a gateway name, for example the domain of your Pexip Infinity deployment plus a "trusted" or "untrusted" label, for example "example.com (trusted)".

4. Select **Generate**.
You are shown the generated access token.



This is the token you must enter into Pexip Infinity.

5. Use the option to **copy the token** to your clipboard.

This is the only time you will be able to see the token before it is stored and encrypted in Google Workspace.

6. Switch to your Pexip Infinity Administrator interface and configure this gateway token in Pexip Infinity:

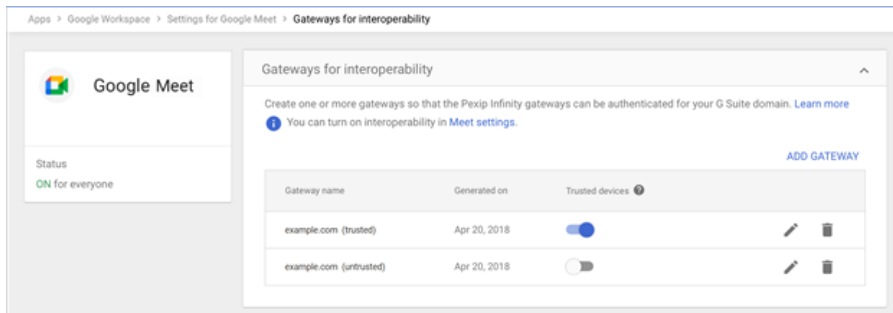
- a. Go to **Call Control > Google Meet Access Tokens** and select **Add Google Meet access token**.
- b. Add the details of your token:

Name	The name that you specify here is used elsewhere in the Pexip Infinity interface when associating the token with a Call Routing Rule or Virtual Reception, so we recommend including an indication if it is your trusted or untrusted token.
Access token	Paste in the access token from your clipboard.

- c. Select **Save**.

7. You can now return to the Google Workspace console and **Close** the token dialog.
8. Use the **Trusted device** button to set the token to either trusted or untrusted as appropriate.
9. If you want to create a trusted and an untrusted token, repeat the above steps to generate the second gateway token.

You can create as many trusted and untrusted gateways as required, although one of each type is normally sufficient. Service providers may need to apply multiple pairs of access tokens for each tenant they are managing.



See [Configuring Pexip Infinity as a Google Meet gateway](#) for more information about Pexip Infinity configuration requirements.

Google Meet interoperability settings

You also need to enable Google Meet interoperability to allow other systems to dial into your Google Meet calls. You do this via **Apps > Google Workspace > Google Meet** and then configure the **Meet video settings** and select the **Interoperability** settings.

Interoperability
Applied at:

Meet interoperability with other video conferencing systems is offered through a partner product, Pexip Infinity Gateway. Before turning this on, perform [these steps](#).

To create gateway access tokens, go to [Gateways for interoperability](#).

☒ **Allow interoperability with other systems**
Google shares with Pexip your domain name plus debugging and troubleshooting information. This includes call logs for meetings that use third-party conferencing systems. Pexip uses this information according to its privacy policy. [Learn more](#)

Meeting ID length

☒ **Use long meeting IDs (preferred)**
Meeting invites will include long meeting IDs that remain valid longer. Third-party video conferencing systems can use calendar integration to get joining information. For systems that support it, users can join meetings without entering meeting IDs.

☐ **Use short meeting IDs**
Meeting invites will link to a web page that lets users get short meeting codes to use to join a meeting.

☐ **Use both long and short meeting IDs**

Gateway IP address
Used by external guests to connect to Meet

198.51.100.40

Gateway hostname

vc.example.com

Meeting ID prefix (only applies to short meeting IDs)

Interoperability with Skype for Business

☒ **Provide an additional gateway for Skype for Business users**

Hostname

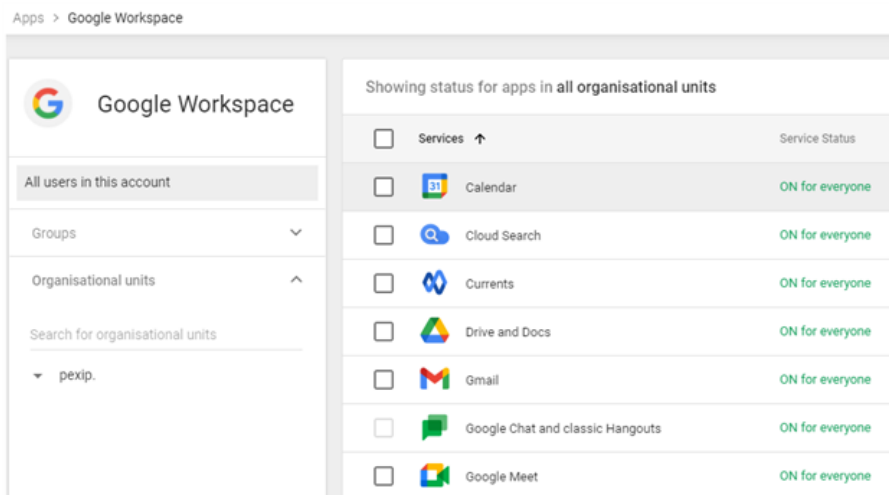
You should configure the following **Interoperability** options:

Name	Description
Allow interoperability with other systems	Select this option to enable gateway interoperability and to configure the other settings.
Meeting ID length	<p>You have three options for setting the length of the meeting ID that is generated by Google:</p> <ul style="list-style-type: none"> Long meeting IDs: this uses a 12 or 13 digit ID. It is added to calendar events and invites when events are created, the join information is easier to find, and it also allows third-party systems to join automatically by SIP URI. Short meeting IDs: this uses a shorter ID with an optional prefix of your choice. Short IDs themselves are not added directly to calendar events and invites, but they are available from a "More joining options" link in the calendar event or invite. Both long and short meeting IDs: provides both long and short IDs. Calendar events and invites show both the long IDs and the link to the page where participants get short IDs. <p>See https://support.google.com/a/answer/7673980 for more information about meeting IDs.</p> <p>We recommend using Long meeting IDs.</p> <p>If you are transitioning from short to long IDs then you can switch immediately to long IDs. Any existing or recurring meetings created with short IDs will still work, but they will continue to only have short IDs. You must ensure that your Call Routing Rules continue to match short and long IDs for the transition period (see this guidance for your regex).</p>

Name	Description
Gateway IP address and Gateway hostname	<p>The IP address and hostname address you specify here are used when Google Meet generates the addresses that allow third-party systems to access the conference. They are used to direct callers to your Pexip Infinity Conferencing Nodes that will then connect them into the Google Meet conference.</p> <ul style="list-style-type: none"> Gateway IP address: set this to the IP address of one of your Conferencing Nodes. Gateway hostname: set this to the name of the DNS SRV record that you have set up for your Conferencing Nodes (e.g. vc.example.com). Alternatively you can use the FQDN of one of your nodes (e.g. px01.vc.example.com). <p>Ensure that you refer to Conferencing Nodes that will be routable from those systems and devices that will be using those dial-in addresses.</p> <p>See DNS record examples for more information about enabling endpoints to route their calls to Conferencing Nodes.</p>
Meeting ID prefix	<p>This prefix applies to short meeting IDs only. If configured, it is prepended to the generated short meeting ID. It can then also be part of any regex-based validation within Pexip Infinity when users are entering meeting IDs.</p> <p>Note that you cannot control the overall length of the generated short meeting ID.</p>
Provide an additional gateway for Skype for Business users	<p>Select this option if you want to generate an address that allows Skype for Business users to join your meetings (available via the "More joining options" link in the calendar event or invite).</p> <p>Leave the Hostname empty.</p>

Controlling access to gateway interoperability

You can enable everybody in your organization to offer gateway interoperability to their Google Meet conferences, or you can limit interoperability to specific OUs (organizational units) or groups. It is not currently possible to enable it for individual users.



Configuring Pexip Infinity as a Google Meet gateway

The Infinity Gateway provides any-to-any video interoperability with Google Meet.

Third-party systems can connect to Google Meet conferences via the Infinity Gateway either by dialing the conference directly or via a Virtual Reception (IVR).

This topic describes the configuration and administrative steps that are required to use Pexip Infinity as a Google Meet gateway. It assumes that you have already performed a basic [installation](#) and configuration of Pexip Infinity. This topic covers:

- [Ensuring a Google Meet \(ghm\) license is installed](#)
- [Configuring your access tokens](#)

- [Configuring Virtual Receptions and Call Routing Rules](#)
 - [Routing directly via the Infinity Gateway](#)
 - [Routing indirectly via a Virtual Reception \(IVR gateway\)](#)
- [DNS and ports requirements](#)
- [Call and participant status](#)
- [Additional deployment information](#)

See [Introduction](#) for more information about the user experience within Google Meet conferences.

Ensuring a Google Meet (ghm) license is installed

You must have a ghm license enabled on your platform (**Platform > Licenses**). This allows you to configure access tokens and route calls to Google Meet.

The ghm license specifies the total number of access tokens you can configure, and is required in addition to the standard Pexip Infinity call licenses. If necessary, contact your Pexip authorized support representative to purchase the required license.

Configuring your access tokens

All communication between Pexip Infinity and Google Meet is authenticated by access tokens that identify your Google Workspace account (see [Generating your gateway access tokens](#)).

To configure your trusted and untrusted access tokens in Pexip Infinity:

1. Go to **Call Control > Google Meet Access Tokens** and select **Add Google Meet access token**.
2. Add the details of your **trusted** token:

Name	The name that you specify here is used elsewhere in the Pexip Infinity interface when associating the token with a Call Routing Rule or Virtual Reception, so we recommend including an indication if it is your trusted or untrusted token.
Access token	The access token value as shown in your Google Workspace account when you generated it.

3. Select **Save**.
4. Repeat the above steps to add the details of your **untrusted** token.

These tokens will now be available to associate with any Virtual Receptions and Call Routing Rules that you configure (as described below) to handle Google Meet conferences.

Note that:

- The access tokens apply to the entire Google Workspace tenant, but you can [enable interoperability](#) on a per-OU (organizational unit) basis within Google Workspace.
- Service providers may need to apply multiple pairs of access tokens for each tenant they are managing.

Configuring Virtual Receptions and Call Routing Rules

There are two ways in which you can configure Pexip Infinity to route calls into Google Meet conferences:

- **Routing directly via the Infinity Gateway:** here you use a Call Routing Rule to route incoming calls for specific alias patterns — that will typically include the meeting ID — directly into the relevant Google Meet conference. This means that the endpoint can dial an alias, such as `1234567890123@example.com` and be taken directly into the conference.
- **Routing indirectly via a Virtual Reception:** here you configure Pexip Infinity to act as an IVR gateway or "lobby" by configuring a Virtual Reception to prompt the caller to enter the meeting ID of the required conference, and then use a Call Routing Rule (typically the same rule as used for direct routing) to route the call into the Google Meet conference. This means that the endpoint can dial a general alias, such as `gmeet@example.com` and then enter the specific meeting ID, such as `1234567890123`, and then be transferred into the conference.

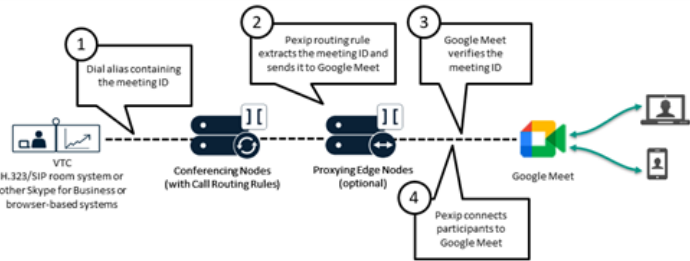
You can use either or both of these two methods, depending upon your requirements. The configuration required for [direct](#) and [indirect](#) routing is explained below.

- i** Depending on your dial plan requirements, you may want to use multiple Call Routing Rules, where some rules use a trusted token and other rules use an untrusted token. For example, if you want to associate calls received via a particular location as trusted, and all calls received in other locations as untrusted then you will need to configure two rules — one rule for calls received in the trusted location that is associated with the trusted token, and then another lower priority rule that is associated with the untrusted token.

Routing directly via the Infinity Gateway

To route calls to Google Meet conferences directly via the Infinity Gateway you need:

- To decide on whether to use long and/or short meeting IDs, and thus the alias pattern that participants will dial to access the Google Meet conferences. See [Configuring Google Workspace for Google Meet integration](#) for more information.
 - The alias pattern will typically include the meeting ID element, for example the pattern could be just <meeting ID> or <meeting ID>@<domain> i.e. the meeting ID and then, optionally, the domain of your Pexip Infinity platform, for example `1234567890123@example.com` to access a Google Meet conference with a meeting ID of 1234567890123.
 - With short meeting IDs you can also configure within Google Workspace a **PIN prefix** (8, for example) to force all meeting IDs to start with that prefix. This may be useful if you have a conflicting dial plan on your video conferencing side that could clash with your Google Meet meeting IDs.
- One or more Call Routing Rules that match that alias pattern and, if necessary, transform it such that it contains just the Google Meet meeting ID which it can then use to connect to the conference. You can use multiple rules to differentiate between devices that are to be treated as trusted or not from Pexip Infinity's perspective, and hence which type of **Access token** is selected and whether **Treat as trusted** is selected or not:
 - If devices register to Pexip Infinity we recommend using two gateway rules: one higher-priority rule to specifically handle registered devices, and one lower-priority rule to handle any device (registered and non-registered).
 - If you are using third-party call control systems you also may want to use different rules to distinguish between calls arriving at Conferencing Nodes in different locations.



To configure each rule:

- Go to **Services > Call Routing** and select **Add Call Routing Rule**.
- Configure the following fields (leave all other fields with default values or as required for your specific deployment):

Option	Description
Name	The name to use to refer to this rule.
Priority	Assign the priority for this rule. If you are creating multiple rules where one rule has Match incoming calls from registered devices only selected, and other rules do not have this option selected, then ensure that the rules that do have Match incoming calls from registered devices only selected have a higher priority (lower number) than those rules where it is not selected (to avoid the call matching first against the "not selected" rule if all of the other rule settings are the same).
Incoming gateway calls	Ensure this option is selected.
Outgoing calls from a conference	Leave unselected.

Option	Description
Calls being handled in location	Applies the rule only if the incoming call is being handled by a Conferencing Node in the selected location. To apply the rule regardless of the location, select <i>Any Location</i> .
Match incoming calls from registered devices only	If devices register to Pexip Infinity, we recommend using two gateway rules: one higher-priority rule to specifically handle registered devices, and one lower-priority rule to handle all devices (registered and non-registered). <ul style="list-style-type: none"> Select this option if you want the rule to apply only to calls received from devices that are registered to Pexip Infinity. Leave this option unselected if you want the rule to apply regardless of whether the device is registered to Pexip Infinity.
Match Infinity Connect (WebRTC / RTMP) Match SIP Match Lync / Skype for Business (MS-SIP) Match H.323	Select one or more of Match Infinity Connect (WebRTC / RTMP) , Match SIP , Match Lync / Skype for Business (MS-SIP) and Match H.323 as appropriate, depending on which types of systems/protocols you want to offer interoperability into Google Meet.
Match against full alias URI	Leave unselected.
Destination alias regex match	Enter a regular expression that will match the calls that are sent to a Google Meet conference. For example, to match an alias in the form <meeting ID> or <meeting ID>@example.com: <ul style="list-style-type: none"> For long meeting IDs, which are 12 or 13 digits, you could use: <code>(^d{12,13})((@example\.com\$))</code> For short meeting IDs, where it has a prefix of 8 and is followed by a further 3-9 digits, you could use: <code>(8d{3,9})((@example\.com\$))</code> If you are transitioning from short IDs to long IDs we recommend that you create two separate rules, rather than defining one rule with a catch-all regex for short and long IDs. These rules should be identical except for the regex matches (using the examples above as a guide), priority (which must be unique for each rule) and name. This means you can more easily monitor if the short ID rule is still being used (the rule name is used as the basis for the service name), and you can simply remove the short ID rule when it is no longer required.
Destination alias regex replace string	If required, enter the regular expression string to transform the originally dialed (matched) alias into the meeting ID to use to place the call into the Google Meet conference. If you do not need to change the alias, leave this field blank. When used with the examples for Destination alias regex match shown above you would use: <code>\1</code> which would extract just the <meeting ID> element of the alias.
Call capability	To support incoming calls via SIP, H.323 and WebRTC/RTMP, <i>Same as incoming call</i> is recommended as this makes the most efficient use of Pexip Infinity resources. If you also want to support calls from Skype for Business / Lync, then you should select <i>Main video + presentation</i> instead to ensure that any SfB/Lync participants are able to escalate from audio-only to a video call after joining the conference (alternatively you can configure a separate rule just for matching incoming calls from Skype for Business / Lync and set that rule to use <i>Main video + presentation</i>).
Theme	If required, assign a customized theme to this rule (which will then be used for callers that use this rule to gateway into Google Meet). For example, the theme could use alternative labels on some of the splash screens that are displayed when connecting to a conference.
Call target	Select <i>Google Meet meeting</i> .



Option	Description
Outgoing location	<p>If required, you can ensure that the outgoing call to the Google Meet conference is handled by a Conferencing Node in a specific location.</p> <p>If an outgoing location is not specified, the call is placed from a Conferencing Node in the same location as the Conferencing Node that is handling the incoming call.</p>
Access token	<p>Select the name of the access token to use to resolve Google Meet IDs. You should select either a trusted or untrusted type of token, depending on whether you want to enable the device to be automatically admitted into the Google Meet conference (subject to also being a trusted endpoint from Pexip Infinity's perspective i.e. if the rule also has Treat as trusted enabled).</p> <p>Typically, you will use a trusted token if Treat as trusted is selected, and an untrusted token if Treat as trusted is not selected.</p>
Treat as trusted	<p>Select Treat as trusted if you want Google Meet to treat the devices routed via this rule as part of the target organization for trust purposes.</p> <p>Typically you will select this option if the rule is handling devices that are trusted from Pexip Infinity's perspective, for example, you could treat the device as trusted if the caller is coming from a specific location, or if the device is registered to Pexip Infinity.</p> <p>This setting is used in conjunction with the Access token setting to control whether the device is automatically admitted into the Google Meet conference.</p>

3. Select **Save**.
4. If you are creating multiple rules, for example when handling whether a device is registered to Pexip Infinity or not, return to step 1 and create the next rule.

Add Call Routing Rule

Name	<input type="text" value="Joining Google Meet"/> *
The name used to refer to this Call Routing Rule. Maximum length: 250 characters.	
Service tag	<input type="text"/>
A unique identifier used to track usage of this Call Routing Rule. For more information, see Tracking usage with a service tag . Maximum length: 250 characters.	
Description	<input type="text"/>
A description of the Call Routing Rule. Maximum length: 250 characters.	
Priority	<input type="text" value="30"/> *
The priority of this rule. Rules are checked in ascending priority order until the first matching rule is found, and it is then applied. Range: 1 to 200.	



Use this rule for...

Incoming gateway calls	<input checked="" type="checkbox"/>	Applies this rule to incoming calls that have not been routed to a Virtual Meeting Room or Virtual Reception, and should be routed via the Pexip Distributed Gateway service .
Outgoing calls from a conference	<input type="checkbox"/>	Applies this rule to outgoing calls placed from a conference service (e.g. when adding a participant to a Virtual Meeting Room) where Automatic routing has been selected. For more information see Configuring Call Routing Rules .
Calls being handled in location	<input type="text" value="Any Location"/>  	Applies the rule only if the incoming call is being handled by a Conferencing Node in the selected location or the outgoing call is being initiated from the selected location. To apply the rule regardless of the location, select Any Location .

When matching incoming Gateway calls...

Match incoming calls from registered devices only	<input checked="" type="checkbox"/>	Only apply this rule to incoming calls from devices, videoconferencing endpoints, soft clients or Infinity Connect clients that are registered to Pexip Infinity. Note that the call must also match one of the selected protocols below. Calls placed from non-registered clients or devices, or from the Infinity Connect Web App will not be routed by this rule if it is enabled.
Match Infinity Connect (WebRTC / RTMP)	<input checked="" type="checkbox"/>	Select whether this rule should apply to incoming calls from Infinity Connect clients (WebRTC / RTMP).
Match SIP	<input checked="" type="checkbox"/>	Select whether this rule should apply to incoming SIP calls.
Match Lync / Skype for Business (MS-SIP)	<input type="checkbox"/>	Select whether this rule should apply to incoming Lync / Skype for Business (MS-SIP) calls.
Match H.323	<input type="checkbox"/>	Select whether this rule should apply to incoming H.323 calls.

Alias match and transform	
Match against full alias URI	<input type="checkbox"/> <p>This setting is for advanced use cases and will not normally be required. By default, Pexip Infinity matches against a parsed version of the destination alias, i.e. it ignores the URI scheme, any other parameters, and any host IP addresses. So, if the original alias is "sip:alice@example.com;transport=tls" for example, then by default the rule will match against "alice@example.com". Select this option to match against the full, unparsed alias instead.</p>
Destination alias regex match	<input type="text" value="(^\\d{12,13})(\$ (@example\\.com\$))"/> * <p>The regular expression that the destination alias (the alias that was dialed) is checked against to see if this rule applies to this call. For help with using regexes, see Regular expression reference. Maximum length: 250 characters.</p>
Destination alias regex replace string	<input type="text" value="\\1"/> <p>The regular expression string used to transform the originally dialed alias (if a match was found). Leave blank to leave the originally dialed alias unchanged. Maximum length: 250 characters.</p>

Call media settings	
Maximum inbound call bandwidth (kbps)	<input type="text"/> <p>This optional field allows you to limit the bandwidth of media being received by Pexip Infinity from each individual participant dialed in via this Call Routing Rule. For more information see Restricting call bandwidth. Range: 128 to 8192.</p>
Maximum outbound call bandwidth (kbps)	<input type="text"/> <p>This optional field allows you to limit the bandwidth of media being sent by Pexip Infinity to each individual participant dialed out from this Call Routing Rule. For more information see Restricting call bandwidth. Range: 128 to 8192.</p>
Call capability	<input type="text" value="Same as incoming call"/> * <p>Maximum media content of the call. The participant being called will not be able to escalate beyond the selected capability. For more information see Controlling media capability.</p>
Maximum call quality	<input type="text" value="Use global setting"/> <p>Sets the maximum call quality for each participant. For more information see Setting and limiting call quality.</p>
Media encryption	<input type="text" value="Use global setting"/> <p>Controls the media encryption requirements for participants connecting to this service. Use global setting: use the global media encryption setting (Platform > Global Settings). Best effort: each participant will use media encryption if their device supports it, otherwise the connection will be unencrypted. Required: all participants must use media encryption. No encryption: all H.323, SIP and MS-SIP participants must use unencrypted media.</p>
Theme	<input type="text" value="Meet"/>   <p>The theme for use with this service. If no theme is selected here, files from the theme that has been selected as the default (Platform configuration > Global settings > Default theme) will be applied. For more information, see Customizing video and voice prompts using themes.</p>

Outgoing call placement	
Call target	<div>Google Meet meeting ▼ *</div> <p>The device or system to which the call is routed. The options are: Registered device or external system: routes the call to a matching registered device if it is currently registered, otherwise attempts to route the call via an external system such as a SIP proxy, Lync / Skype for Business server, H.323 gatekeeper or other gateway/ITSP. Registered devices only: routes the call to a matching registered device only (providing it is currently registered). Lync / Skype for Business meeting direct (Conference ID in dialed alias): routes the call via a Lync / Skype for Business server to a Lync / Skype for Business meeting. Note that the destination alias must be transformed into just a Lync / Skype for Business Conference ID. Lync / Skype for Business clients, or meetings via a Virtual Reception: routes the call via a Lync / Skype for Business server either to a Lync / Skype for Business client, or - for calls that have come via a Virtual Reception - to a Lync / Skype for Business meeting. For Lync / Skype for Business meetings via Virtual Reception routing, ensure that Match against full alias URI is selected and that the Destination alias regex match ends with .* Google Meet meeting: routes the call to a Google Meet meeting.</p>
Outgoing location	<div>Automatic ▼ ✎ +</div> <p>When calling an external system, this forces the outgoing call to be handled by a Conferencing Node in a specific location. When calling a Lync / Skype for Business meeting, a Conferencing Node in this location will handle the outgoing call, and - for Lync / Skype for Business meeting direct targets - perform the Conference ID lookup on the Lync / Skype for Business server. Select Automatic to allow Pexip Infinity to automatically select which Conferencing Node to use.</p>
TURN server	<div>----- ▼ ✎ +</div> <p>The TURN server to be used for outbound Lync / Skype for Business (MS-SIP) calls (where applicable). For more information, see About TURN servers.</p>
STUN server	<div>----- ▼ ✎ +</div> <p>The STUN server to be used for outbound Lync / Skype for Business (MS-SIP) calls (where applicable).</p>
Access token	<div>example.com trusted token ▼ ✎ +</div> <p>The access token to use to resolve Google Hangouts Meet meeting codes.</p>
Treat as trusted	<input checked="" type="checkbox"/> <p>This indicates the target of this routing rule will treat the caller as part of the target organization for trust purposes.</p>
Rule state	
Enable this rule	<input checked="" type="checkbox"/> <p>Determines if the rule is enabled or not. Any disabled rules still appear in the rules list but are ignored. Use this setting to test configuration changes, or to temporarily disable specific rules.</p>

Save

Save and add another

?

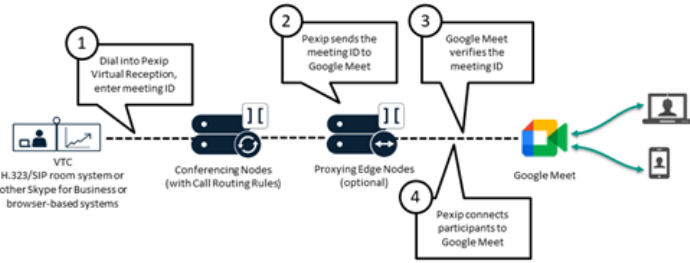
Using the direct gateway service

After the Call Routing Rule has been configured, third-party systems and devices can now dial an alias that matches your specified pattern (e.g. 1234567890123 or 1234567890123@example.com) to be routed directly into the appropriate Google Meet conference (in this example the conference with a meeting ID of 1234567890123).

Routing indirectly via a Virtual Reception (IVR gateway)

To route calls to Google Meet conferences via a Virtual Reception (IVR gateway) you need:

- A Virtual Reception configured specifically to handle Google Meet conferences.
- A Call Routing Rule to route the calls handled by the Virtual Reception into the relevant Google Meet conference. Typically you would configure the Virtual Reception and Call Routing Rule patterns so that the same rule can also support direct routing as described above.



The Virtual Reception requests the caller to enter the Google Meet meeting ID which it then sends to Google Meet for verification. You can then optionally transform the meeting ID to meet any dial plan requirements, before the Infinity Gateway then matches the (optionally transformed) meeting ID and routes the caller to the appropriate Google Meet conference.



To configure the Virtual Reception:

1. Go to **Services > Virtual Receptions** and select **Add Virtual Reception**.
2. Configure the following fields (leave all other fields with default values or as required for your specific deployment):




Option	Description
Name	The name to use to refer to this Virtual Reception, for example "Google Meet IVR gateway".
Theme	If required, assign a customized theme to this Virtual Reception to brand it as the gateway to Google Meet conferences, for example by customizing the voice prompts or changing the appearance of the Virtual Reception splash screen.
Virtual Reception type	Select Google Meet .
Access token	Select the name of the access token to use to resolve Google Meet IDs. When configuring a Virtual Reception it does not matter if you use a trusted or untrusted access token.
Lookup location	Specify the location that contains the Conferencing Nodes (typically Proxying Edge Nodes) that will perform the service lookup (meeting ID verification) on Google Meet.
Post-lookup regex match	<p>This is an optional field. It is typically used in conjunction with the Post-lookup regex replace string to transform the meeting ID entered into the Virtual Reception into a distinct alias pattern that will match a Call Routing Rule that is configured to route calls into Google Meet conferences.</p> <p>Note that this match and transform occurs after the meeting ID entered by the user has been sent to Google Meet for verification.</p> <p>In most cases, you would typically set the regex match to:</p> <p><code>(.*)</code></p> <p>which matches everything entered into the Virtual Reception.</p>
Post lookup regex replace string	<p>This may be used in conjunction with the Post-lookup regex match field to transform the meeting ID entered by the caller.</p> <p>In our example, setting it to:</p> <p><code>\1</code></p> <p>will simply pass on the entered meeting ID unchanged. This will then typically work in conjunction with the Call Routing Rule that you configured above for direct routing and then directs the call to Google Meet.</p>
Alias	Enter the alias that users will dial to use this Virtual Reception to place calls into Google Meet conferences, for example <code>gmeet@example.com</code> .

3. Select **Save**.

Add Virtual Reception

Name	<input type="text" value="Google Meet IVR"/> *
The name used to refer to this Virtual Reception. Maximum length: 250 characters.	
Description	<input type="text"/>
A description of the Virtual Reception. Maximum length: 250 characters.	
Theme	<input type="text" value="Google Meet"/>  
The theme for use with this service. If no theme is selected here, files from the theme that has been selected as the default (Platform configuration > Global settings > Default theme) will be applied. For more information, see Customizing video and voice prompts using themes .	

Service options

Virtual Reception type	<input type="text" value="Google Meet"/> *
The type of this Virtual Reception. Select Lync / Skype for Business if this Virtual Reception is to act as an IVR gateway to scheduled and ad hoc Lync / Skype for Business meetings. Select Google Meet if this Virtual Reception is to act as an IVR gateway to Google Meet meetings. Otherwise, select Regular.	
Access token	<input type="text" value="example.com trusted token"/>  
The access token to use to resolve Google Hangouts Meet meeting codes.	
Lookup location	<input type="text" value="-----"/>  
If selected, a Conferencing Node in this system location will perform the service lookup. If a location is not selected, the IVR ingress node will perform the lookup.	
Post-lookup regex match	<input type="text" value="(.)"/>
An optional regular expression used to match against the meeting code, after the service lookup has been performed. Maximum length: 250 characters.	
Post-lookup regex replace string	<input type="text" value="\1"/>
An optional regular expression used to transform the meeting code so that, for example, it will match a Call Routing Rule for onward routing to the required conference. (Only applies if a Post-lookup regex match is also configured and the meeting code matches that regex.) Maximum length: 250 characters.	

Advanced options (Show)

Aliases

Alias: #1	
Alias	<input type="text" value="gmeet@example.com"/> *
The dial string used to join this service, in the form that it will be received by Pexip Infinity. This alias must include any domain that is automatically added by the participant's endpoint or call control system, or dialed by the participant. For more information, see About aliases . Maximum length: 250 characters.	
Description	<input type="text"/>
An optional description of the alias. Note that this description may be displayed to end users on registered Infinity Connect clients who are performing a directory search. Maximum length: 250 characters.	

[Add another Alias](#)[Save](#)[Save and add another](#)[?](#)

To configure the associated Call Routing Rule:

- Configure the Call Routing Rule as described above for [direct routing](#).
- If you want to use a different rule for routing via a Virtual Reception than the rule you are using for direct routing (e.g. because you want to limit the supported incoming call protocols, or use a different outgoing location for calls placed via the Virtual Reception), then follow the same principles as the direct routing rule, but use a different alias pattern in your Virtual Reception's Post-lookup regex replace string and your rule's Destination alias regex match string.

Using the Google Meet IVR gateway service

After the Virtual Reception and Call Routing Rule have been configured, third-party systems and devices can now dial the alias of the Virtual Reception (e.g. `gmeet@example.com`) and then, when prompted by the IVR service, enter the meeting ID of the Google Meet conference they want to join.

The Infinity Gateway will then route the call into the appropriate Google Meet conference.

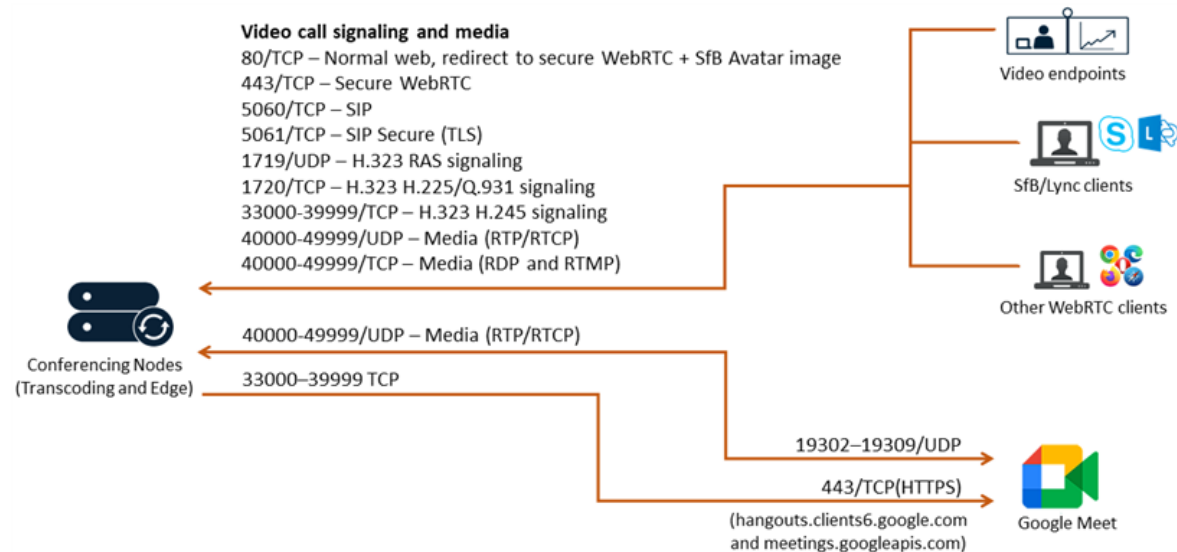
DNS and ports requirements

You need to ensure that the endpoints and systems you want to gateway into Google Meet can call into Pexip Infinity Conferencing Nodes, and that Conferencing Nodes can call out to Google Meet.

These are the port usage rules for call signaling and media between Google Meet and Conferencing Nodes (Proxying Edge Nodes and Transcoding Conferencing Nodes):

Source address	Source port	Destination address	Dest. port	Protocol	Notes
Google Meet	19302–19309	Conferencing Node	40000–49999 **	UDP	SRTP/SRTCP
Conferencing Node	33000–39999 **	Google Meet (<code>hangouts.clients6.google.com</code> and <code>meetings.googleapis.com</code>)	443	TCP (HTTPS)	
Conferencing Node	40000–49999 **	Google Meet	19302–19309	UDP	SRTP/SRTCP

** Configurable via the Media port range start/end, and Signaling port range start/end options (see [About global settings](#)).



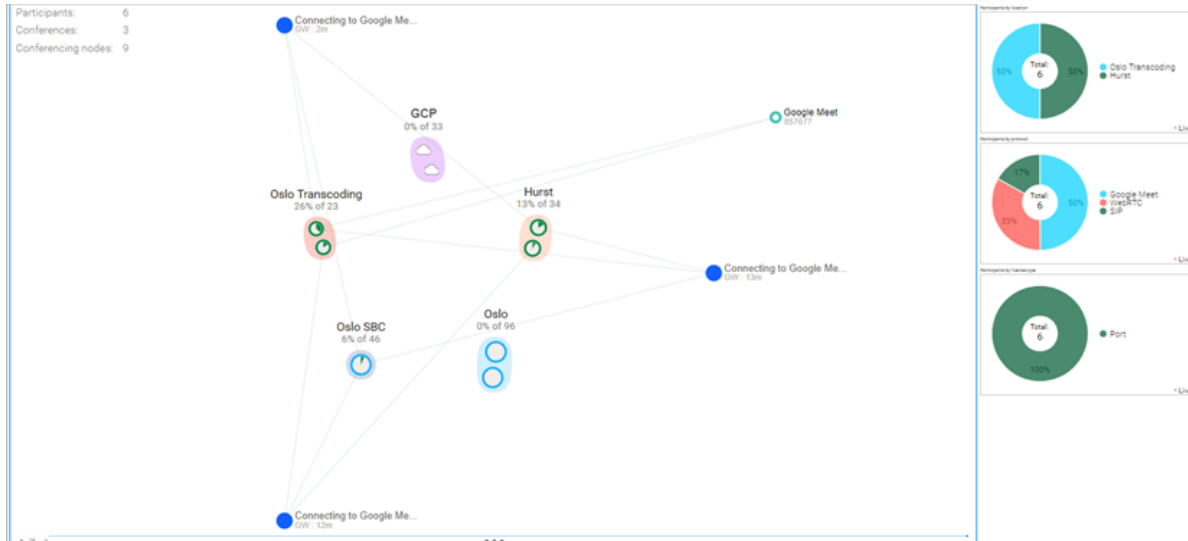
Call signaling and media ports

- See [DNS record examples](#) for information about enabling endpoints to route their calls to Conferencing Nodes.
- See [Pexip Infinity port usage and firewall guidance](#) for complete information about the ports used when Conferencing Nodes connect to Google Meet and other devices.

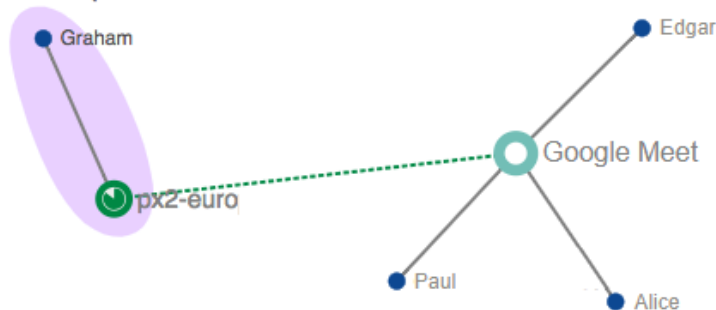
Call and participant status

When using the Pexip Infinity Administrator interface to monitor calls that are placed into Google Meet conferences, you should note that:

- Each participant who is gatewayed into a Google Meet conference is listed as a separate gateway call. However, if multiple participants are connected to the same Google Meet conference, the Live View (Status > Live View) shows them as connected to the same external conference.



GCP-Europe



- When viewing the participant status for a gateway call, the meeting ID, such as 1234567890123, is shown as a participant alias. This participant represents the gateway call leg into Google Meet. If you look at the media streams associated with that participant you see that:
 - Pexip Infinity sends (subject to bandwidth) three VP8 video streams (each at different resolutions) and one 1 audio stream to Google Meet for that participant.
 - Pexip Infinity receives one video and one audio stream for each external participant in the conference, up to a maximum of 8 video streams (to support Pexip's standard 1+7 layout). If there are more than 8 other participants then only an audio stream is received for those extra participants.

Media streams

Type	Start time	Node	Tx codec	Tx bitrate (kbps)	Tx resolution	Tx framerate	Tx packets sent	Tx packets lost	Tx current packet loss	Tx jitter (ms)	Rx codec	Rx bitrate (kbps)	Rx resolution	Rx framerate	Rx packets received	Rx packets lost	Rx current packet loss	Rx jitter (ms)
Video	2018-06-25 13:43:32 (BST)	10.47.2.21	VP8	129	320x180	30.0	2403	1	0.0%	33.3	Off	0		0.0	0	0	0.0%	0.0
Video	2018-06-25 13:43:32 (BST)	10.47.2.21	Off	0		0.0	0	0	0.0%	0.0	VP8	2516	1280x720	30.0	18911	0	0.0%	0.8
Video	2018-06-25 13:43:32 (BST)	10.47.2.21	Off	0		0.0	0	0	0.0%	0.0	VP8	128	320x180	25.0	1222	0	0.0%	2.3
Video	2018-06-25 13:43:32 (BST)	10.47.2.21	VP8	303	640x360	30.0	3474	2	0.0%	33.3	Off	0		0.0	0	0	0.0%	0.0
Video	2018-06-25 13:43:32 (BST)	10.47.2.21	VP8	1266	1280x720	30.0	10612	0	0.0%	33.3	Off	0		0.0	0	0	0.0%	0.0
Audio	2018-06-25 13:43:32 (BST)	10.47.2.21	Opus	64			3987	5	0.0%	0.9	Off	0			0	0	0.0%	0.0
Audio	2018-06-25 13:43:32 (BST)	10.47.2.21	Off	0			0	0	0.0%	0.0	Opus	1			2118	1	0.0%	1.2
Audio	2018-06-25 13:43:32 (BST)	10.47.2.21	Off	0			0	0	0.0%	0.0	Opus	0			446	0	0.0%	3.1

8 Media streams

Other participant aliases that are displayed for that call include the device that placed the call (such as name@example.com) and one or more aliases in the format spaces/<id>/devices/<id> which represent the other participants in the Google Meet conference.

- You cannot control (e.g. disconnect, mute or transfer) any of the other participants connected to the Google Meet conference.

Additional deployment information

- Each participant who is gatewayed via Pexip Infinity into a Google Meet conference consumes two call licenses (one for the inbound leg of the call and one for the outbound leg, as is standard for calls via the Infinity Gateway calls). Any external participants who are connected directly to the Google Meet conference do not consume a license. See [Pexip Infinity license installation and usage](#) for more information.
- You cannot limit the **Maximum outbound call bandwidth** (the call leg towards Google Meet) — it is fixed at 2 Mbps.
- If the Google Meet conference is recorded, "streaming enabled" indicators are included in the video stream sent to gatewayed participants.
- Chat messages are supported in both directions between Google Meet and other chat-enabled clients. However, the name of the sender from the Google Meet side is not identified on messages received by Skype for Business clients.

Integrating Microsoft Teams with Pexip Infinity

For the most recent information, see [Integrating Microsoft Teams with Pexip Infinity](#).

Integrating Epic telehealth with Pexip Infinity

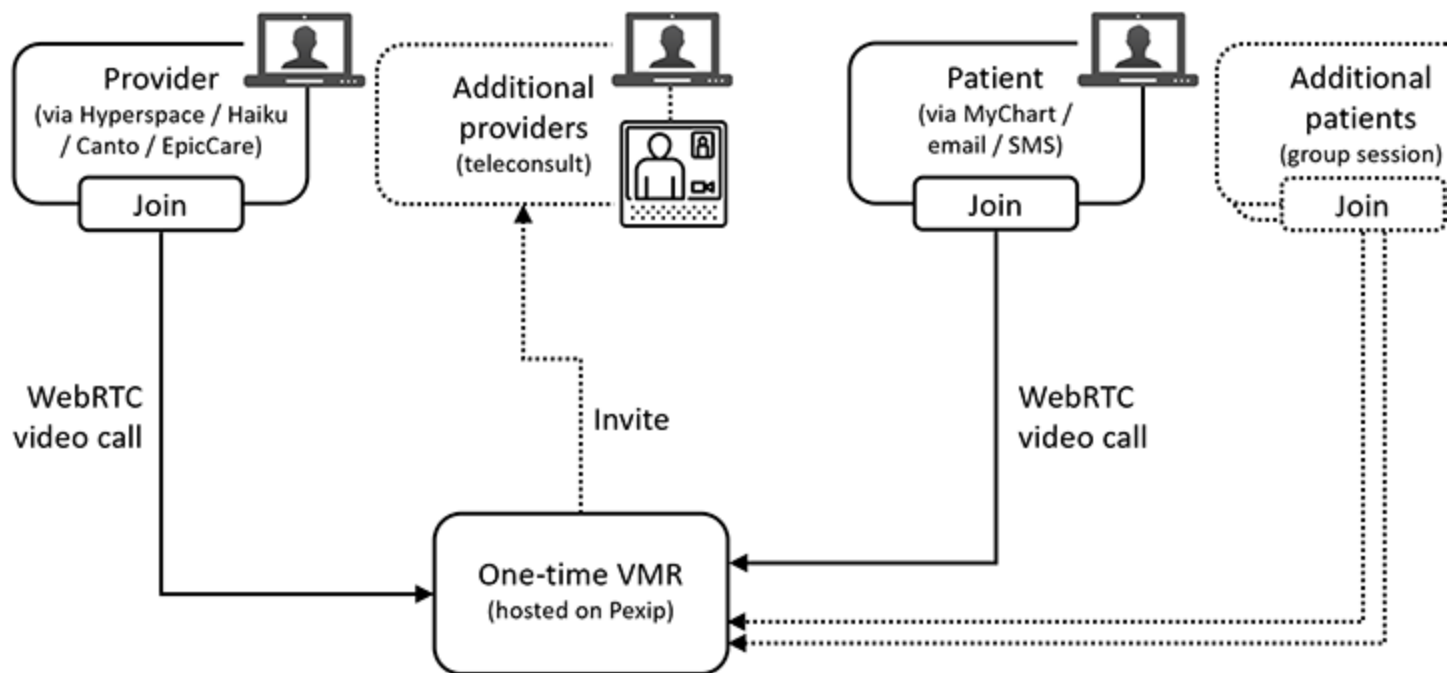
Epic telehealth integration with Pexip Infinity

Pexip's Epic telehealth integration enables healthcare organizations to hold video-based visits in Pexip Virtual Meeting Room (VMRs). The Pexip solution:

- Provides secure, easy-to-join telehealth visits from any location.
- Integrates seamlessly with Epic's standard video visit workflow.
- Uses context-aware linking within an Epic appointment to enable providers and patients to meet together over video.
- Allows providers (physicians/doctors) to directly launch their video visit from Hyperspace, Haiku, Canto or EpicCare.
- Allows patients to launch the video visit directly from MyChart, or via join links sent by email or SMS text messages.
- Supports clinic-to-clinic and teleconsult workflows (remote provider).
- Requires no downloads or plugins.
- Enables HIPAA compliance.

Epic Electronic Health Record (EHR) customers include hospitals, health systems, and physician practices. For more information, see Pexip's listing in the [Epic App Orchard](#).

Each video visit is held in a one-time VMR within Pexip Infinity via the WebRTC-based Infinity Connect web app. Typically the visit will involve a single provider and patient, but multiple providers could join the same video visit — invited either by Epic-based workflows, or by calling out directly from the Pexip VMR. Multiple patients can also join the same video visit if it is a group session.



How it works

When it is time to go to their appointment, the patient clicks a button in MyChart, or clicks a join link sent to them by email or SMS text message, and this launches a video visit browser session (using Infinity Connect).

The provider (doctor) can see the patient's appointment in Epic and may also be notified that the patient has connected and is ready to be seen. The provider clicks a button in their Epic system to launch their video visit and join the session.

The Pexip Infinity implementation and join process works as follows:

- Each Epic appointment is held in a one-time VMR within Pexip Infinity:
 - When the provider presses the Join button in their Epic app, an Infinity Connect web app call is placed via the Infinity Gateway into the one-time VMR. The join URL is configured to take them straight into the VMR.
 - Similarly when the patient presses their Join button or clicks their join link, they also launch a web app call directly into the same one-time VMR.
 - Epic generates a unique, one-time-use, join URL for each participant (provider and patient) for every join attempt. Each URL contains a unique Pexip Infinity alias, which is derived from the appointment information.
 - If a call fails to connect for whatever reason, or the user gets disconnected, they can rejoin the same appointment but they must not try to re-use the same join URL (as it will always fail on subsequent re-use). They must close the browser tab, go back to their Epic healthcare application (typically their particular hospital's Mychart portal) and re-launch the call, or click again on their email/SMS join link. They will end up in the same meeting as before, just via a different alias.
 - Within Pexip Infinity, each alias for the same appointment is associated with the same Pexip Infinity service name (which is also derived from the appointment information). This ensures that each join attempt for the same appointment is taken to the same one-time VMR.
 - Thus, multiple providers and patients all meet in the same VMR if they all share the same appointment.
- Providers are treated as Hosts and patients are treated as Guests. The one-time VMR has a Host PIN, but no Guest PIN.
- Providers (Hosts) can dial out from the VMR and invite other participants if required.
- If patients (Guests) join the VMR before a provider (Host) has connected, they are held at the **Waiting for the host** screen until a provider joins (who automatically opens the conference with the Host PIN included within their join URL).
- Each video visit launches into an external browser session so as to allow the user continued access to either Hyperspace or MyChart.
- The launching of the external Infinity Connect web application from the various Epic platforms uses SMART on FHIR OAuth 2.0 authentication (a set of open specifications to integrate apps with Electronic Health Records, portals, Health Information Exchanges, and other health IT systems). When a provider/patient clicks "join" to launch the Pexip video session they may get challenged by OAuth to re-enter their Epic sign-on credentials. This is purely down to timing and is not in Pexip's control.

Outline of the integration process

Here is an overview of the integration process, including the steps and interactions taken between the customer, Pexip and Epic:

1. The customer deploys a standard Pexip Infinity platform in their self-hosted environment.
2. The customer obtains and applies a telehealth license from Pexip (in addition to any other Pexip Infinity licenses they need).
3. The customer performs a basic (non-telehealth) test call to ensure that at least one Conferencing Node is reachable from the Internet, has proper certificates, and that call connectivity is working correctly (e.g. by calling two people into a test VMR via the Infinity Connect web app).
4. The customer formally requests the Pexip integration app via [Epic App Orchard](#).
5. A Pexip administrator approves the customer request and securely stores the production client secret and non-production client secret.
6. Pexip informs the customer of the production and non-production client secrets via secure means. The customer is then responsible for storing them securely and entering them into the Pexip Infinity Administrator interface.
7. Pexip also informs the customer of the patient and provider application client IDs, and the backend OAuth2 application client ID that are appropriate to their environment.
8. The customer configures their Epic FDI record and Epic telehealth profile to their Pexip Infinity deployment. This is best performed simultaneously as there is some data to be shared between the two systems. The customer should:
 - Create an Epic FDI record, and generate their encryption key and securely share it with Epic.
 - Add an Epic telehealth profile to their Pexip Infinity deployment, using the appropriate client secrets, application client IDs and encryption key settings. The profile's uuid identifier should be included in the corresponding CRYPTURL in the Epic FDI record (see [Creating an Epic telehealth profile](#)).
9. The customer sets up the private/public keypairs to support OAuth2 authentication to Epic for patients joining via email/SMS.
10. The customer provides Pexip with:
 - The patient OAuth2 redirect URL and provider OAuth2 redirect URL that they have configured on their deployment.
 - The public key files they created to support OAuth2 authentication for patients joining via email/SMS.

11. Pexip asks Epic to add the two OAuth2 redirect URLs to the patient and provider applications on the Epic side, to add the public key files, and to synchronize the client secrets and encryption keys.
12. When the changes are made and have propagated on the Epic backend and on Pexip Infinity, the system is ready for testing and validation.
13. Pexip permanently and securely destroys any customer keys in our possession — secure storage of these (e.g. for backup or restore purposes) is now the customer's responsibility.

More information

For full details on the mandatory and optional integration configuration steps, and further reference information, see:

- [Configuring Pexip Infinity to integrate with Epic telehealth](#)
- [Optional features and customizations for Epic telehealth integrations](#)
- [Monitoring, maintenance and reference information for Epic telehealth integrations](#)
- [Troubleshooting and call setup information for Epic telehealth integrations](#)

Configuring Pexip Infinity to integrate with Epic telehealth

This topic describes the mandatory configuration and administrative steps that are required to integrate Pexip Infinity with an Epic telehealth system. This topic covers:

- [Prerequisites and integration basics](#)
- [Platform routing and firewall requirements](#)
- [Ensuring a telehealth integration license is installed](#)
- [Creating an Epic telehealth profile](#)
- [Creating a Call Routing Rule for incoming calls](#)
- [Configuring Pexip Infinity to integrate with Epic telehealth](#)
- [Optional features](#)

Prerequisites and integration basics

Before you perform the telehealth integration steps described here, you should have:

- Already performed a basic [installation](#) and configuration of Pexip Infinity. Any of Pexip Infinity's deployment models may be used (on-premises, cloud-hosted etc). Note that if you are using Pexip Smart Scale (PSS), then you need to deploy and configure a web proxy in order to use PSS with Epic telehealth.
- Confirmed that (non-telehealth) call connectivity is working correctly e.g. by calling two people into a test VMR via the Infinity Connect web app.

Summary of Pexip Infinity configuration steps

After you have completed your standard implementation of the Pexip Infinity platform, these are the specific Epic telehealth integration configuration steps that you must perform within Pexip Infinity:

1. [Install a telehealth integration license](#) (Platform > Licenses).
2. [Create an Epic telehealth profile](#) (Call Control > Epic Telehealth Profile).
3. [Create a Call Routing Rule](#) to manage and route the incoming telehealth calls from providers and patients (Services > Call Routing).

All of these steps are described in detail below.

You can also perform some [optional configuration](#) to customize the integration or use additional features.

Recommended Epic crypto algorithms and compatibility

The recommended crypto algorithms for different Epic versions are:

- AES-256-CBC is recommended for Epic versions from August 2019 onwards
- AES-128-CBC may be used for older Epic versions (August 2018 onwards)

The oldest Epic version that works with Pexip Infinity is August 2018.

Epic configuration requirements

As well as the Pexip Infinity configuration guidelines contained within this documentation, you must also refer to Epic's *Pexip Implementation Guide*.

The configuration of the Epic and the Pexip Infinity installations need to align precisely (with certain shared secret configuration values) for the integration to work correctly. Please note the following important configuration requirements that must be implemented within Epic:

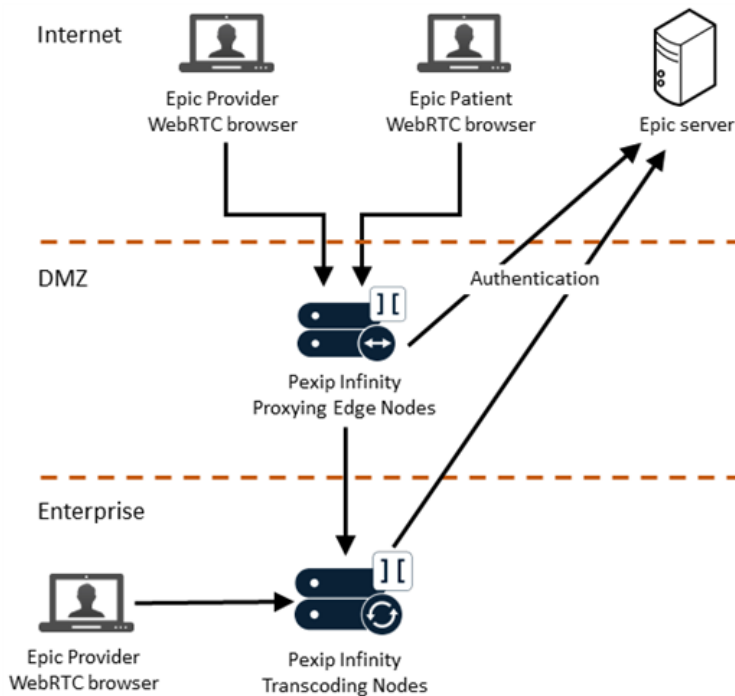
- FDI records must be created in Epic to integrate Epic with the Pexip Infinity installation. These ensure that both Epic and Pexip Infinity use the same algorithm and key to pass information between themselves.
- The uuid identifier of the **Epic telehealth profile** on Pexip Infinity must be included in the corresponding Launch URL (CRYPTURL) in the Epic FDI record.
- The OAuth2 redirect URLs defined in the **Epic telehealth profile** on Pexip Infinity must be registered within the Epic configuration.
- Refer to Epic/Citrix support if there is a need to launch a browser in a Hyperspace/Citrix environment to ensure that Chrome is used for the video call instead of Internet Explorer.

Platform routing and firewall requirements

Each Epic customer requires a self-hosted Pexip Infinity platform with the following routing/firewall requirements:

- At least one Conferencing Node (which could be a Proxying Edge Node) must be accessible from the computers/devices running the Epic apps, which typically means that the node needs to be publicly reachable. Alternatively, access could be routed via a reverse proxy. If required, the jinja templates can be used to load balance traffic across multiple Conferencing Nodes or to split provider traffic and patient traffic between different pools of Conferencing Nodes. For large environments (100+ concurrent video calls) we recommend that you contact your Pexip authorized support representative to discuss your architectural design.
 - Firewall and DNS requirements are as for a standard Pexip Infinity public deployment. However, note that:
 - All inbound calls are placed via the Infinity Connect web app (WebRTC), so no SIP or H.323 ports are required for incoming calls.
 - All Conferencing Nodes that may be involved in handling telehealth calls, including all "internal" transcoding nodes that are processing conference media, must be able to connect out to port 443/TCP (HTTPS) on the Epic provider and patient systems.
 - If device pairing is used, or you want the ability to call out from the VMR to invite other participants into the meeting, then you need to ensure that you allow SIP or H.323 call signaling and media from your Conferencing Nodes to those video endpoints.
 - There is nothing from the Epic/public side of the integration that needs to access the Pexip Infinity Management Node.
- i** The web entry point used for the OAuth2 redirect URLs (typically a Proxying Edge Node or reverse proxy) must be stable. Subsequent changes to DNS names will require hard-to-coordinate changes on Epic's side. Ensure that you use a DNS name that is suitable over the long term.

A typical deployment environment may look like this:



Ensuring a telehealth integration license is installed

You must have a telehealth integration license enabled on your Pexip Infinity platform (**Platform > Licenses**).

The **telehealth** license enables you to configure an Epic telehealth profile, and is required in addition to the standard Pexip Infinity call licenses. If necessary, contact your Pexip authorized support representative to purchase the required license.

Creating an Epic telehealth profile

The Epic telehealth profile configured on Pexip Infinity defines how the join URLs for providers and patients are constructed, and includes the shared encryption settings and keys that allow Pexip Infinity and the Epic servers to communicate securely.


- i** At the start of the integration process, your Pexip authorized support representative will have given you various configuration items, including customer-specific secret credentials. Be very careful about copying and pasting the Epic encryption key and Application client secrets into your Epic telehealth profile. If there are any extra (or missing) characters in these fields, telehealth calls will fail to launch. Furthermore, as these fields are stored encrypted and are not decryptable by Pexip support, this can make it difficult to discover any typos when troubleshooting.

Go to **Call Control > Epic Telehealth Profile**, select **Add Telehealth Profile** and configure the profile as described:

Option	Description
Name	The name of the telehealth profile. This name appears in the web app participant list (roster) and in the Pexip Infinity administrator interface as part of the service name, so we recommend setting it to the name of the hospital or healthcare organization. Example: Healthcare Org
Description	A description of the telehealth profile.

Option	Description
Unique uuid identifier for this telehealth profile	<p>A unique identifier for the telehealth profile. A value is automatically assigned and there is normally no need to modify it.</p> <p>Example: 12daaebc-ea75-43c1-a6c7-c044be66a36e</p> <p>The corresponding Launch URL (CRYPTURL) in the Epic FDI record should include this uuid, for example:</p> <p>https://yourpexipinfinityserver.example.com/api/telehealth/v1/patient/oauth2smartlaunch/12daaebc-ea75-43c1-a6c7-c044be66a36e?<parameters></p> <p>with the <parameters> replaced with appropriate parameters for your deployment as per Epic's own Pexip integration guide documentation. Note that the CRYPTURL configured in the Epic FDI record determines which telehealth profile is used for a particular call.</p>
Domain name	The name of the domain to use for telehealth aliases associated with this telehealth profile location.
Base URL of the Epic server	<p>The base HTTPS URL of the Epic server.</p> <p>Example: https://epic.example.com/</p>
Epic OAuth2 base URL	<p>The base HTTPS URL of the Epic server's OAuth2 APIs.</p> <p>Example: https://epic.example.com/interconnect-aocurprd-oauth</p>
Patient OAuth2 redirect URL	<p>The OAuth2 redirect URL for the patient telehealth application.</p> <p>Format: https://[infinity deployment]/api/telehealth/v1/patient/oauth2authorized/[uuid]</p> <p>where:</p> <ul style="list-style-type: none"> [infinity deployment] is the address of your public Conferencing Node, Proxying Edge Node, reverse proxy or load balancer. [uuid] is a unique uuid identifier for this telehealth profile (typically this should be the same as the uuid identifier entered above for this profile). <p>Example: https://yourinfinitydeployment.example.com/api/telehealth/v1/patient/oauth2authorized/12daaebc-ea75-43c1-a6c7-c044be66a36e</p>
Provider OAuth2 redirect URL	<p>The OAuth2 redirect URL for the provider telehealth application.</p> <p>Format: https://[infinity deployment]/api/telehealth/v1/provider/oauth2authorized/[uuid]</p> <p>where:</p> <ul style="list-style-type: none"> [infinity deployment] is the address of your public Conferencing Node, Proxying Edge Node, reverse proxy or load balancer. [uuid] is a unique uuid identifier for this telehealth profile (typically this should be the same as the uuid identifier entered above for this profile). <p>Example: https://yourinfinitydeployment.example.com/api/telehealth/v1/provider/oauth2authorized/12daaebc-ea75-43c1-a6c7-c044be66a36e</p>
Infinity web application base URL	<p>The public base URL for the Pexip Infinity web app.</p> <p>Example: https://[infinity deployment]/</p> <p>where:</p> <ul style="list-style-type: none"> [infinity deployment] is the address of your public Conferencing Node, Proxying Edge Node, reverse proxy or load balancer. <p>Example: https://yourinfinitydeployment.example.com/</p>
Patient application Client ID	The unique OAuth2 application Client ID assigned by Epic for the Pexip telehealth patient application to use when contacting clients.
Provider application Client ID	The unique OAuth2 application Client ID assigned by Epic for the Pexip telehealth provider application to use when contacting clients.

Option	Description
Template for provider aliases	<p>The Jinja2 template used for generating provider aliases used by providers (e.g. doctors) when joining a telehealth conference. This must include the value of <code>{{base_telehealth_alias}}</code> somewhere in the alias, although you can use your own choice of prefix and/or suffix.</p> <p>Default: <code>telehealth.{{base_telehealth_alias}}@{{telehealth_integration.telehealth_call_domain}}</code></p> <p>See Supported Jinja2 template variables for the full set of template variables that can be used in provider alias templates.</p>
Template for patient aliases	<p>The Jinja2 template used for generating patient aliases used by patients when joining a telehealth conference. This must include the value of <code>{{base_telehealth_alias}}</code> somewhere in the alias, although you can use your own choice of prefix and/or suffix.</p> <p>Default: <code>telehealth.{{base_telehealth_alias}}@{{telehealth_integration.telehealth_call_domain}}</code></p>
Provider WebRTC join link template	<p>The Jinja2 template used for generating HTTPS web join links used by providers (e.g. doctors) when joining a telehealth conference. It must include:</p> <ul style="list-style-type: none"> <code>{{telehealth_alias}}</code> variable somewhere in the link (usually as a <code>conference=</code> parameter) <code>pin</code> parameter (so that the provider automatically opens the conference when they join) <p>and you may customize other aspects of the URI if required.</p> <p>Default: <code>{{telehealth_integration.infinity_webapp_server_base_url}}/webapp/#/?conference={{telehealth_alias}}&pin={{pin}}&name={{display_name}}</code></p> <p>See Supported Jinja2 template variables for the full set of template variables that can be used in join link templates.</p>
Patient WebRTC join link template	<p>The Jinja2 template used for generating HTTPS web join links used by patients when joining a telehealth conference. It must include:</p> <ul style="list-style-type: none"> <code>{{telehealth_alias}}</code> variable somewhere in the link (usually as a <code>conference=</code> parameter) <p>and you may customize other aspects of the URI if required, although you should not include the <code>pin</code> parameter.</p> <p>Default: <code>{{telehealth_integration.infinity_webapp_server_base_url}}/webapp/#/?conference={{telehealth_alias}}&name={{display_name}}&role=guest</code></p>
Service name template	<p>The Jinja2 template used for generating the telehealth conference/appointment name. It must include:</p> <ul style="list-style-type: none"> <code>{{unique_encounter_id}}</code> variable somewhere in the generated name; however you may add your own prefix or suffix to the ID if required. <p>i The same service name is used for all participants who join the conference. Therefore if this integration is used for group appointments, the template must not include <code>{{launch_information.fname}}</code> or <code>{{launch_information.lname}}</code> as these will differ from patient to patient — and in which case you must modify the default template content.</p> <p>Default: <code>Appointment for {{launch_information.fname}} {{launch_information.lname}} {{telehealth_integration.name}}:{{unique_encounter_id}}{% if launch_information.encfacnpiname %} in {{launch_information.encfacnpiname}}{% endif %}</code></p> <p>This default template generates a name of "Appointment for <first name> <last name> <profile name>:<Epic appointment ID>" and also adds "in <encounter department>" if the department information has been provided by Epic.</p> <p>See Supported Jinja2 template variables for the full set of template variables that can be used in the service name template.</p>
Error page template for launch failures	<p>The Jinja2 template used for generating the error page shown to users if the telehealth call launch fails. You may adapt the template styles and text as appropriate for your environment.</p> <p>See Error page template for launch failures for more information.</p>

Option	Description
Epic encryption key	<p>The encryption key used to encrypt and decrypt telehealth parameters passed from Epic to Pexip Infinity when launching calls associated with this telehealth Integration.</p> <ul style="list-style-type: none"> If the encryption algorithm is AES-256-CBC then the key type must be base64-encoded key (not password), and the key itself must be a base64-encoded 32 byte random value. If the encryption algorithm is AES-128-CBC then the key type may be base64-encoded key or password. If a base 64 key is used the key itself must be a base64-encoded 16 byte random value. A password may be any length. <p> Both Epic and Pexip Infinity must be configured to use exactly the same algorithm and key in order for information to be passed correctly.</p> <p>See Epic encryption keys, client secrets and OAuth2 keypairs for more information.</p>
Encryption key type	<p>The type of the encryption key. It may be a simple text password from which a key will be derived, or it may be a predefined base64-encoded key. If AES-256-CBC is the encryption algorithm used, a base64-encoded key is mandated. Options are:</p> <ul style="list-style-type: none"> Simple password Base 64 encoded key <p>Default: <i>Simple password</i></p>
Epic encryption algorithm	<p>The encryption algorithm used by Epic when encrypting telehealth call parameters during telehealth call launch. For Epic releases from August 2019 onwards AES-256-CBC is recommended. Options are:</p> <ul style="list-style-type: none"> AES-128-CBC AES-256-CBC <p>Default: <i>AES-128-CBC</i></p>
Patient application Client Secret	<p>The unique OAuth2 application Client ID assigned by Epic for the Pexip telehealth patient application.</p> <p>Currently, the same value is used for both the Patient and Provider secrets on a single telehealth profile.</p> <p>See Epic encryption keys, client secrets and OAuth2 keypairs for more information.</p>
Provider application Client Secret	<p>The unique OAuth2 application Client ID assigned by Epic for the Pexip telehealth provider application. This should be set to the same value as the Patient application Client Secret.</p>
Email and SMS helper application	
Backend OAuth2 application Client ID	<p>The unique OAuth2 application Client ID assigned by Epic for the Pexip backend application to use when contacting the Epic server.</p> <p>See Backend OAuth2 client ID and private/public keypairs for more information.</p>
Epic OAuth2 token URL for the backend application	<p>The HTTPS URL of the Epic server's OAuth2 token URL.</p> <p>Example: https://epic.example.com/interconnect-aocurprd-oauth/oauth2/token</p>
Backend OAuth2 private key	<p>The private key for the Pexip backend OAuth2 application to use when authenticating to Epic.</p> <p>See Backend OAuth2 client ID and private/public keypairs for more information.</p>

Supported jinja2 template variables

The following table shows which variables may be used in which templates. A ✓ indicates that the variable **may** be included in the template, whereas indicates that the variable **must** be present in the template.

Variable name	Description	Allowed in template?			
		Alias	Join link	Service name	Error page
base_telehealth_alias	<p>A unique one-time alias for this join attempt, that must be included in the alias templates. It forms part of the overall conference alias that is specified in the WebRTC join link.</p> <p>This is the element that the associated Call Routing Rule should extract from the dialed conference alias when routing the incoming call.</p>				
detailed_debug_information	Contains debug information about the failure to launch a call that can be used to help diagnose the problem.				✓
display_name	<p>The name of the provider/patient displayed in the VMR:</p> <ul style="list-style-type: none"> Providers: the telehealth_integration.name (e.g. "Healthcare Org") as the name of the practitioner may not be available to Pexip Infinity. Patients: the display name supplied by Epic during launch. 		✓		
error_name	<p>The error name (returned from Epic) that can be used to decide which messages to display to the user on the error page:</p> <ul style="list-style-type: none"> An error_name of "BorvoGetTelehealthDirectLaunchTokenError" is returned if the launch link was not valid. This error could be expected to occur in a working environment. It might be because the user has selected a link for an old appointment, or for an appointment that isn't ready to start yet. Any other error names that might be returned are the result of an unexpected error, and we recommend that these are all treated in the same manner. 				✓
launch_information	<p>This variable contains appointment information provided by Epic.</p> <p>It may only be used in the service name template. It contains the following fields:</p> <ul style="list-style-type: none"> fname: patient first name lname: patient last name encfacnpiname: encounter department <p>Example: "Appointment for {{launch_information.fname}} {{launch_information.lname}} in {{launch_information.encfacnpiname}}"</p>			✓	
pin	<p>A randomly-generated 8-digit Host PIN.</p> <p>This is mandatory for provider join links, and optional for patient join links.</p>				
telehealth_alias	The provider/patient alias (generated from the provider and patient alias templates).				
telehealth_integration	<p>This variable contains various properties of the telehealth profile configured in Pexip Infinity. It contains the following fields:</p> <ul style="list-style-type: none"> telehealth_call_domain: the profile's Domain name name: the profile's Name description: the profile's Description integration_uuid: the profile's uuid identifier infinity_webapp_server_base_url: The profile's Infinity web application base URL <p>Example: "telehealth.{{base_telehealth_alias}}@{{telehealth_integration.telehealth_call_domain}}"</p>	✓	✓	✓	

Variable name	Description	Allowed in template?			
		Alias	Join link	Service name	Error page
telehealth_request_id	The telehealth call identifier. This is unique for every attempt to join an appointment.	✓	✓		✓
unique_encounter_id	A unique appointment ID provided by Epic. This is the same for all participants (providers and patients) joining a given appointment (or "encounter" in Epic terminology). It must be referenced in the service name template.	✓	✓		
utc_time	The current time in UTC, used to indicate when a call launch error occurred.				✓
* Mandatory.					

See [Jinja2 templates and filters](#) for more information about using jinja templates.

Error page template for launch failures

An error page is displayed to users if a telehealth call fails to launch correctly.

The contents of the page is determined by the **Error page template for launch failures** field in the telehealth profile. This field is an HTML/jinja2 template that typically contains a mixture of HTML markup and styles, literal text and jinja2 variables. It can also contain jinja2 control structures that allow you to vary the content of the page based upon certain conditions.

The default template is shown below but you may adapt the content as appropriate for your environment:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Something went wrong</title>
    <style>
      .pexip-cell {
        padding: 20px;
      }
      .main-title {
        font-size: 22px;
        color: #4a4a4a;
      }
      .pexip-heading {
        font-weight: bold;
        font-size: 14px;
        color: #4a4a4a;
      }
      .pexip-info {
        font-size: 12px;
        color: #4990e2;
      }
    </style>
  </head>
  <body>
    <br>
    <table style=
      "width: 100%; border-collapse: collapse; border: 1px solid #e9e9e9; font-family: Calibri, sans-serif, serif;">
      <tbody>
        <tr>
          <td colspan="2" class="pexip-cell" style="vertical-align: top;">
            {% if error_name == "BorvoGetTelehealthDirectLaunchTokenError" %}
            <p><span class="main-title">That link isn't working at the moment. It might be a link for an old appointment, or for an appointment
that isn't ready to start yet.</span></p>
            <p><span class="pexip-heading">Please check that you are using the correct link for your appointment today.</span></p>
            <p>
              <span class="pexip-heading">Extra information:</span><br>
              <span class="pexip-info">telehealth_request_id: {{telehealth_request_id}}</span><br>
              {% if detailed_debug_information %}
              <span class="pexip-info">Detailed debugging information: {{detailed_debug_information}}</span>
              {% endif %}
            </p>
          </td>
        </tr>
      </tbody>
    </table>
```



```

        {% else %}
        <p><span class="main-title">Something went wrong - looks like an error on our end or on the network.</span></p>
        <p><span class="pexip-heading">Please close this window and try reconnecting.</span></p>
        <p>
        <span class="pexip-heading">If the error persists, please share the information below with your provider for
        troubleshooting:</span><br>
        <span class="pexip-info">utc_time: {{utc_time}}</span><br>
        <span class="pexip-info">telehealth_request_id: {{telehealth_request_id}}</span><br>
        {% if detailed_debug_information %}
        <span class="pexip-info">Detailed debugging information: {{detailed_debug_information}}</span>
        {% endif %}
        </p>
        {% endif %}
    </td>
</tr>
</tbody>
</table>
</body>
</html>

```

The default template has the following logic:

- First, It defines some styles for use within the page.
- It tests if `error_name == "BorvoGetTelehealthDirectLaunchTokenError"`:
 - If true, it displays a message that the user may have used a link that is out of date or is not ready yet.
 - Otherwise (`error_name` is not "BorvoGetTelehealthDirectLaunchTokenError") this means an unexpected error has occurred and an appropriate message is displayed ("Something went wrong - looks like an error on our end or on the network").
- In both cases, the `telehealth_request_id` is displayed, and if `detailed_debug_information` is available then that is also displayed on the page.

The following variables are available in the error page template for launch failures: `detailed_debug_information`, `error_name`, `telehealth_request_id` and `utc_time` (full details are included [above](#)).

Creating a Call Routing Rule for incoming calls

You must create a Call Routing Rule within Pexip Infinity to manage and route the incoming telehealth calls appropriately.

1. Go to **Services > Call Routing** and select **Add Call Routing Rule**.
2. Configure the following fields (leave all other fields with default values or as required for your specific deployment):

Option	Description
Name	The name of this rule e.g. "Epic telehealth".
Priority	Assign the priority for this rule. If you are creating multiple rules you need to ensure that any other rules with a higher priority (lower number) will not process your telehealth calls.
Incoming gateway calls	Ensure this option is selected.
Outgoing calls from a conference	Leave this option unselected.
Calls being handled in location	Applies the rule only if the incoming call is being handled by a Conferencing Node in the selected location. To apply the rule regardless of the location, select Any Location .
Match incoming calls from registered devices only	Leave this option unselected.

Option	Description
Match Infinity Connect (WebRTC / RTMP)	Select Match Infinity Connect (WebRTC / RTMP) .
Match SIP	Do not select any of the other protocols.
Match Lync / Skype for Business (MS-SIP)	
Match H.323	
Match against full alias URI	Leave this option unselected.
Destination alias regex match	<p>The objective of the match/replace strings is to extract the <code>base_telehealth_alias</code> from the dialed alias in the incoming call received by Pexip Infinity from Epic.</p> <p>The match string should map to the provider/patient alias templates in the telehealth profile. The example regex match of <code>telehealth\.(.*)@(.*)</code> works perfectly with the default telehealth profile alias templates.</p> <p>Example: <code>telehealth\.(.*)@(.*)</code></p>
Destination alias regex replace string	<p>When used with the example Destination alias regex match shown above you would use:</p> <p><code>\1</code></p> <p>which would extract the <code>base_telehealth_alias</code> from the dialed alias.</p>
Theme	If required, assign a customized theme to this rule. For example, the theme could use alternative labels on some of the splash screens that are displayed when connecting the call. See Optional features and customizations for Epic telehealth integrations for more information.
Call target	Select Epic Telehealth meeting .

3. Select **Save**.

Optional features

In addition to the basic configuration requirements described here, there are extra optional features you may want to consider that require additional configuration, including:

- **Device pairing:** this allows a provider to pair the Infinity Connect web app or desktop client with an external SIP or H.323 device and use it to handle the audio/video aspects of the call.
- **Branding and customization:** you can apply a range of branding or customization options:
 - Create your own theme to customize the splash screens (such as the **Waiting for the host** screen).
 - Create a customized Infinity Connect web app to control the web app's appearance and behavior on call completion.
 - Use local policy to change the default layout and other properties of the VMR.
- **Dial out (teleconsult):** enable dialing out to another provider from within the conference.

See [Optional features and customizations for Epic telehealth integrations](#) for more information.

Optional features and customizations for Epic telehealth integrations

This topic describes some optional features, customization and branding options for Epic telehealth integrations with Pexip Infinity:

- [Customizing the audio prompts and splash screens via themes](#)
- [Changing the branding and styling of the Infinity Connect web app](#)
- [Advanced Infinity Connect web app customization via plugins](#)
- [Customizing the behavior at the end of a call](#)
- [Changing the disconnect timeout](#)
- [Overriding the one-time VMR layout and configuration via local policy scripts](#)
- [Pairing the Infinity Connect client with a video device](#)

- [Calling out from a VMR to invite other consultants](#)
- [Using the Infinity Connect desktop client](#)

Customizing the audio prompts and splash screens via themes

The audio prompts that are played out to patients and providers, and the splash screens that are displayed prior to joining the meeting, can all be customized by creating and uploading an alternative theme.

In particular you may want to customize the **Waiting for the host** screen that is displayed to patients while they are waiting for the provider to join.

We have also provided an alternative `conf-waithostpin_48kHz_mono.wav` file where instead of the standard "Waiting for the conference Host to join. If you are the conference Host, please enter the conference PIN number now" message, the alternative file says "Waiting for your healthcare provider to start the appointment". You can download this alternative theme file as `healthcare.zip` from <https://dl.pexip.com/resources/themes/index.html>.

You can either set a new default theme (for all calls on the Pexip Infinity platform), or apply the theme to the Call Routing Rule handling the incoming telehealth calls.

See [Customizing conference images and voice prompts using themes](#) for more information.

Changing the branding and styling of the Infinity Connect web app

The look and feel of the Infinity Connect web app can be customized, for example to apply language translations or to change the color scheme for buttons, icons and other graphic indicators on the home page of the web app. (Note that the in-call elements, such as the splash screens, are customized via themes instead).

You can use the Pexip branding portal to apply a whole range of customizations — see [Customizing the Infinity Connect clients](#) for more information.

Advanced Infinity Connect web app customization via plugins

If you want to offer dynamic control of the one-time VMR to the provider (VMR Host) so that, for example, they can change the layout for group sessions then you can develop a bespoke Infinity Connect plugin.

Other potential uses for plugins could be to facilitate access to a translation call center service.

Customizing the behavior at the end of a call

By default, when a user finishes a call placed via the Infinity Connect web app they are returned to the app home page. However, you can change this behavior so that they are sent to a different URL/webpage instead, such as:

- The healthcare organization's home page.
- A custom webpage that says something like "This call has ended. Please close your browser page."

We recommend doing this as it ensures that the patient is taken away from the web app when the call disconnects or if the call drops. This is particularly important as the URLs generated when a patient or provider first joins a meeting are single use. Thus it's not possible to rejoin a meeting using the old alias — the user has to go back into the Epic healthcare app (e.g. MyChart) and rejoin from there.

Note that you cannot apply this type of customization via the branding portal. This redirect behavior can only be configured via manual customization of the Infinity Connect web app, by defining a `disconnectDestination` in the `settings.json` file. However, you can first use the branding portal to apply your other customization requirements and then add the `disconnectDestination` option to the files generated by the branding portal before uploading the branding package to the Management Node.

Changing the disconnect timeout

By default, guests (e.g. patients) who are waiting for hosts (e.g. healthcare providers) are automatically disconnected after 900 seconds (15 minutes). If you'd like a longer or shorter timeout, configure **Waiting for Host timeout** as appropriate in **Platform > Global Settings > Service Configuration**.

Overriding the one-time VMR layout and configuration via local policy scripts

The one-time VMR that is generated for each telehealth call has the following properties:

- 1+7 layout
- Display names enabled
- Host PIN only
- Guests can present
- Chat: as per global settings (**Platform > Global Settings > Connectivity**, then deselect or select **Enable chat**.)
- Uses the theme defined in the Call Routing Rule handling the incoming call.

If you want to use a different layout or disable overlay names for every telehealth call, you can configure a local policy script to override the default configuration. Local policy uses the Jinja2 scripting language and allows you to control Pexip Infinity's call behavior.

When using local policy, the call information available to policy includes a `telehealth_request_id` field. This field is only present in telehealth calls and identifies each individual call or join attempt. You could also identify telehealth calls based on the `local_alias` (dialed alias) pattern.

Pairing the Infinity Connect client with a video device

The Infinity Connect "device pairing" feature allows a provider to pair the Infinity Connect web app or desktop client with an external SIP or H.323 device and use it to handle the audio/video aspects of the call. When pairing is configured:

- The provider joins the call in the usual way and when the Infinity Connect client joins, another call is automatically placed out to the paired device and that device is brought into the conference.
- The provider can use the desktop video device, and still control the call and share content from their computer.

When providers set up the paired device:

- Each individual provider has to perform a one-time configuration step to associate their paired device with their Infinity Connect (by entering the video address of their device). This has to be set up before they join a telehealth meeting.
- We recommend that the provider disables **Showing presentation on this device** — this means that any presentation is shown on the PC / web app beside the video device and the doctor can also present from the web app if necessary.

You must configure Pexip Infinity with an additional "outgoing" Call Routing Rule to ensure that the call out to the paired device is routed correctly (you only need one rule that will work for all providers and their paired devices):

1. Go to **Services > Call Routing** and select **Add Call Routing Rule**.
2. Configure the following fields (leave all other fields with default values or as required for your specific deployment):

Option	Description
Name	The name used to refer to this rule e.g. "Telehealth device pairing".
Priority	Assign the priority for this rule. If you are creating multiple rules you need to ensure that any other rules with a higher priority (lower number) will not process these outgoing calls.
Incoming gateway calls	Leave this option unselected.
Outgoing calls from a conference	Ensure this option is selected.
Destination alias regex match	This is a regular expression that must match the destination alias i.e. the video address of the paired device. Note that the regex must match the entire alias — a partial match is treated as a non-match. See Regular expression (regex) reference for information about writing regular expressions.
Call target	The device or system to which the call is routed. The suitable options are: <ul style="list-style-type: none"> ◦ Registered device or external system: route the call to a matching registered device if it is currently registered, otherwise attempt to route the call via an external system such as a SIP proxy or H.323 gatekeeper. ◦ Registered devices only: route the call to a matching registered device only (providing it is currently registered).

Option	Description
Protocol	The protocol used to place the outgoing call. Select <i>SIP</i> or <i>H.323</i> as appropriate for the type of device you need to call. Note that if the call is to a registered device, Pexip Infinity will instead use the protocol that the device used to make the registration.
SIP Proxy	You can optionally specify the SIP Proxy to use to place an outgoing SIP call, otherwise select <i>Use DNS</i> .
H.323 Gatekeeper	You can optionally specify the H.323 Gatekeeper to use to place an outgoing H.323 call, otherwise select <i>Use DNS</i> .

3. Select **Save**.

Calling out from a VMR to invite other consultants

A provider who is already within a video call can use the Infinity Connect client to call out directly to other providers (if appropriate, instead of using Epic's remote consultation order procedures).

The dial-out works in a similar manner to device pairing as described above, but in this case the provider always has to manually enter the video address (alias) of the person they want to invite into the conference.

This process also requires a Call Routing Rule to be configured within Pexip Infinity to handle the outgoing call. It can use exactly the same rule as described above for device pairing. The only thing you need to ensure is that the **Destination alias regex match** is appropriate to match the video addresses of the remote providers you may want to invite into the video visit.

See [Dialing out via the Infinity Connect client](#) for more information.

Using the Infinity Connect desktop client

Some providers may want to use the Infinity Connect desktop client instead of the web app.

In this case the **Provider WebRTC join link template** in the Epic telehealth profile should be set to: **pexip://{{telehealth_alias}}?pin={{pin}}**

This URL structure causes the telehealth call to be placed via the Infinity Connect desktop client (which must already be installed on the provider's computer).

We recommend that patients continue to use the Infinity Connect web app.

Monitoring, maintenance and reference information for Epic telehealth integrations

This topic describes some monitoring options and additional reference information for Epic telehealth integrations with Pexip Infinity:

- [Monitoring appointments via Pexip Infinity](#)
- [Appointment and participant names](#)
- [Epic encryption keys, client secrets and OAuth2 keypairs](#)
- [Access and use of data shared between Epic and Pexip Infinity](#)
- [Removing a Pexip/Epic integration](#)

For troubleshooting information, see [Troubleshooting and call setup information for Epic telehealth integrations](#).

Monitoring appointments via Pexip Infinity

Active and historic appointments (conferences) can be viewed via the Pexip Infinity Administrator interface.

Each appointment is represented as a single conference within the **Status** pages on Pexip Infinity, and each conference typically has two participants — the provider and the patient. However, there could be additional participants if it is a group session or an additional provider is added to the call.

The appointment and participant names are constructed as described below.

Appointment and participant names

Appointment and participant names are displayed within the call itself (in the Infinity Connect roster) to all of the participants, and they are also shown within the Pexip Infinity Administrator interface status pages.

By default the appointment name (referred to as the service or conference name within Pexip Infinity) name takes the form: "Appointment for <patient name> <telehealth profile name>:<Epic unique appointment id> in <encounter department>" but you can control this by configuring the **Service name template** in the Pexip Infinity [telehealth profile](#).

Note that "in <encounter department>" is only included if the department is known. For example:

- "Appointment for Beverley P Spinder Healthcare Org:6ea6d1427f684721abd21c1634386f7b6 in Radiology" or
- "Appointment for Kai Oosman Picardo Healthcare Org:5ec8f1667f531781afe19c1774322f7b8"

The participants list contains:

- The **Name** field of the telehealth profile for the provider e.g. "Healthcare Org".
- The first name / middle name / last name for the patient e.g. "Beverley P Spinder".

Epic encryption keys, client secrets and OAuth2 keypairs

This section contains more information about how to complete some of the Pexip Infinity telehealth profile settings concerning encryption keys, client secrets and backend OAuth2 keypairs.

Epic encryption key

The **Epic encryption key** is a secret used for encrypting some of the information passed from Epic to Pexip. It should be a cryptographically strong random value. It is used in the **Epic telehealth profile** within Pexip Infinity and FDI records within Epic.

To generate a secret for use with Epic AES-256-CBC crypto (used with Epic version August 2019 onwards), you must generate a random 32 byte, base64 encoded key. (If you are using an older version of Epic, you can use either a 16 byte key generated using a similar method to that described below, or a simple password.)

You can generate the key using any tools that Epic may provide, or by using a Pexip command line tool as described below:

1. Connect over ssh into the Management Node as user admin with the appropriate password.
2. Issue the following command:

```
pextelehealthkeygen 32
```

Example output:

```
value: fTW2Kw1JZ6JEpRS+R1N2fruKPgggBe+Y9txLIk3QpA0=
```

The **value** output to the screen is a random 32 byte key, generated by OpenSSL, that is encoded into human-readable text using base64 encoding.

We recommend that you store this value securely (for example, using your corporate password database) in case you need to recover or reinstall your Pexip system without a suitable backup being available.

Client Secrets

The **Provider application Client Secret** and **Patient application Client Secret** that must be configured in the **Epic telehealth profile** are generated by Epic during the signup process and passed to Pexip, who will relay them securely to the end customer. Currently, the same value is used for both the provider and patient secrets on a single Epic telehealth profile.

A production and non-production variant of each secret is generated. These should be treated as being as sensitive as passwords, and you should ensure that the production and non-production variants are noted distinguishably.

Backend OAuth2 client ID and private/public keypairs

This section explains the procedure for configuring the **Email and SMS helper application** settings in an Epic telehealth profile, to support patients joining via links sent by email or SMS text messages.

Your Pexip representative will share the **Backend OAuth2 application Client ID** with you. In common with other application secrets that make up an integration, your production and non-production Epic environments use different Client IDs.

Each client ID needs its own associated unique public/private keypair, so you need to perform the following procedure twice to generate a keypair:

1. Log in to the Pexip Infinity Management Node over SSH.
2. Run the following commands:

```
cd /dev/shm
openssl genrsa -out ./privatekey.pem 3072
openssl req -new -x509 -key ./privatekey.pem -out ./publickey509.pem -subj '/CN=PexipBackendDirectLaunchApp'
```

This generates two files: **privatekey.pem** and **publickey509.pem**.

Make sure that the key length is at least 2048 bits (our instructions above specify 3072 bits).

3. Copy **privatekey.pem** and **publickey509.pem** off the Management Node using an SCP (Secure Copy) client, for example WinSCP.
4. Delete the files from the Management Node by running the following commands:

```
rm ./privatekey.pem
rm ./publickey509.pem
```

5. Make a secure backup of the **privatekey.pem** and **publickey509.pem** files (you will need them if you need to reinstall your Pexip system).

Make sure to indicate clearly if the keys are intended for your production environment or your non-production environment. You may want to rename the generated files to distinguish between the files you want to use for your production and non-production environments, and to ensure you do not overwrite the first set of files with the second set of files when you repeat this process.

i **privatekey.pem** is sensitive — so ensure you store it in a manner that complies with your organization's secret storage security policy (e.g. in a secure location such as your corporate password database).

6. Repeat the process to generate a second pair of keys (one pair for your production environment and one pair for your non-production environment).

You can now complete your configuration:

1. Configure your Pexip Infinity Epic telehealth profile for your production environment:
 - a. In a text editor, open the **privatekey.pem** file you want to use for your production environment.
 - b. Copy/paste the contents into the **Backend OAuth2 private key** field in the telehealth profile.
 - c. Check you have also entered the production **Backend OAuth2 application Client ID** that Pexip shared with you, and select **Save**.
2. Repeat the step above for your non-production Pexip Infinity environment, but this time using the non-production **Backend OAuth2 application Client ID** and the other **privatekey.pem** file.
3. Send your two **publickey509.pem** files via email to your Pexip authorized support representative who will relay them to Epic so they can add the keys to their central system. Make sure to indicate clearly which key is for your production environment, and which is for your non-production environment.

i Only send the public keys. Do not send the private keys.

Access and use of data shared between Epic and Pexip Infinity

Please refer to Epic's documentation ("Integration Record Setup" in their *Pexip Implementation Guide*), which is available from your Epic support representative, for information about the data passed from Epic to Pexip Infinity.

The Pexip Infinity deployment obtains data from Epic that is necessary to launch the appointment in Pexip Infinity. This typically includes the first, middle and last name of the patient, the username of the launching user, the department encounter name e.g. radiology, and the CSN (contact serial number — an encrypted ID that identifies an appointment). The same data is also used for providers.

Pexip Infinity writes data to Epic. In particular it informs Epic when telehealth appointment calls terminate.

Privacy and security

The Pexip Infinity self-hosted solution supports the industry standards for communication encryption for end-user devices, ensuring that all video calls and shared media content is secure and kept private even if it crosses the internet. The entire deployment and all its data, including call status, diagnostic logs and call history, is completely under the ownership and control of the enterprise, even when deployed in the cloud.

Removing a Pexip/Epic integration

If you no longer require Epic telehealth integration, you must remove the specific Epic-related configuration from your Pexip Infinity deployment:

- The Epic telehealth profile (**Call Control > Epic Telehealth Profile**).
- The Call Routing Rule that manages the incoming telehealth calls (**Services > Call Routing**).
- Remove the telehealth integration license from your platform (**Platform > Licenses**).

You should also remove any optional customizations you may have applied, including:

- Any branding or customization overrides:
 - Any themes that customized the splash screens (**Services > Themes**).
 - Any customized Infinity Connect web app settings (**Services > Web App Customization**).
 - Any local policy to customize the VMR (**Call Control > Policy Profiles**).
- Any additional Call Routing Rules that manage device pairing and/or dial out (teleconsult) from the conference (**Services > Call Routing**).

Please talk to your Epic provider for information about how to remove your Pexip integration from the Epic system itself.

Troubleshooting and call setup information for Epic telehealth integrations

This topic provides some guidance on troubleshooting installation and call launching issues with telehealth integrations with Pexip Infinity, and it also contains some detailed [call setup](#) information.

Installation and call launching issues

During initial integration it is possible that various misconfigurations may cause authentication with Epic to fail when trying to make any telehealth calls. If authentication fails, the system will usually display a screen containing a short textual error message, and the telehealth call will not attempt to launch. As the failures here are potentially security related, this error page intentionally does not give too much away, but there is usually a short error reason presented, as described below.

"TelehealthIntegrationNotFoundException Error"

If this occurs, check that the Epic server is configured to use a launch URL that is correctly formatted and that contains the exact same (unique to your deployment) uuid as configured in the telehealth profile's **Unique uuid identifier for this telehealth profile** field.

"MetadataFetchError"

This error generally indicates some sort of network related problem (such as connectivity or DNS resolution issues). It occurs when Pexip Infinity was unable to access an unauthenticated (open) API exposed by the Epic server that provides Pexip Infinity with metadata pertaining to the Epic deployment.

Verify that the Epic server configured in the telehealth profile's **Base URL of the Epic server** and **Epic OAUTH2 base URL** fields is operational, reachable, and that it can be resolved using the DNS settings configured in all system locations (including both edge and transcoding locations as appropriate), and is not being blocked by a firewall.

If a web proxy is configured in any location, verify that it too is reachable (and, if applicable, its DNS name is resolvable using the DNS server in the relevant location).

"Oauth2 Launch Error"

If this error occurs, check that:

- The Epic server is passing non-blank "launch=" and "iss=" parameters as part of the initial launch URL.
- The **Epic encryption key** was entered correctly in the telehealth profile, and that **Encryption key type** and **Epic encryption algorithm** are both configured appropriately to match what is configured on the Epic side.
- The **Patient application Client ID** and **Provider application Client ID** are appropriately configured in the telehealth profile for this deployment, and that you are using the production or non-production values as appropriate to match what is configured on the Epic side.
- The **Patient application Client Secret** and **Provider application Client Secret** were both entered correctly in the telehealth profile for this deployment, and that you are using the production or non-production values as appropriate.
- Pexip Infinity's access to the Epic server is not being blocked by a firewall.

"Epic OAuth2 Error"

This Epic-generated message can occur if the **OAuth2 redirect URIs** are not configured within Epic or there is a mismatch between what is configured on the Epic and Pexip sides.

Further investigation

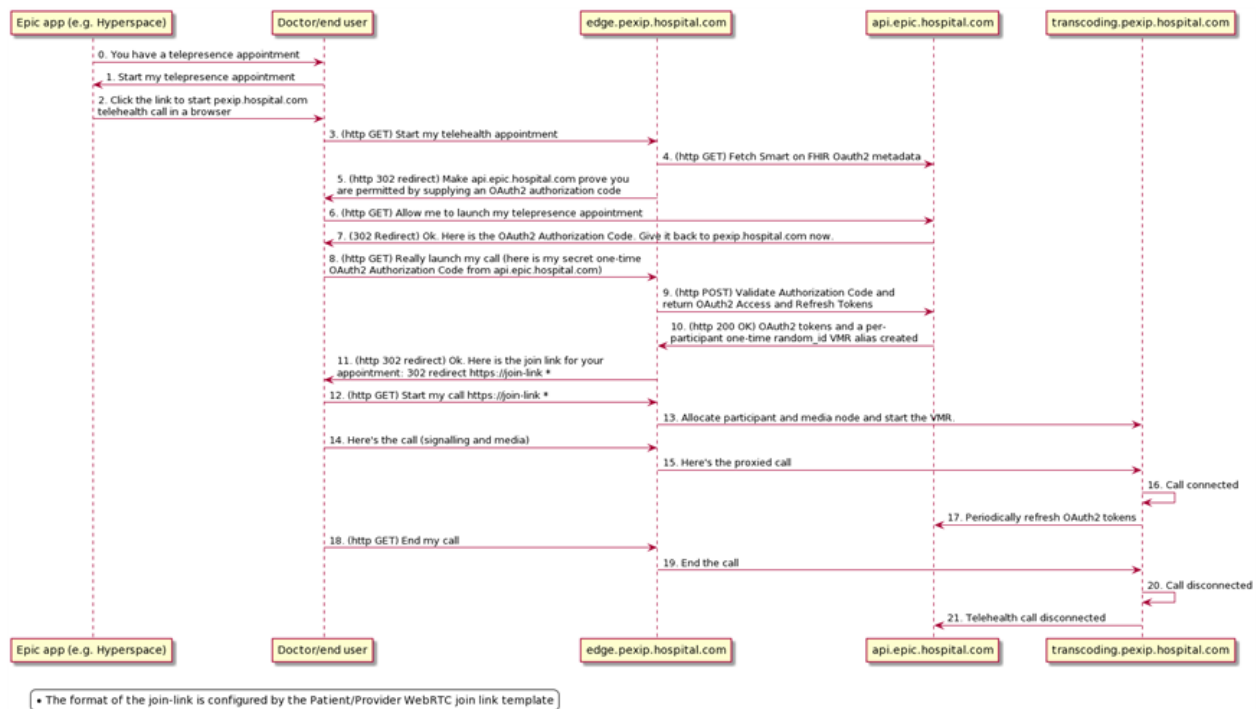
If none of the above steps helps solve your initial launch integration problems, your Pexip authorized support representative may ask you to enable extra debug tracing, reproduce the failing call scenario and then afterwards to download a diagnostic snapshot.

Detailed call setup information

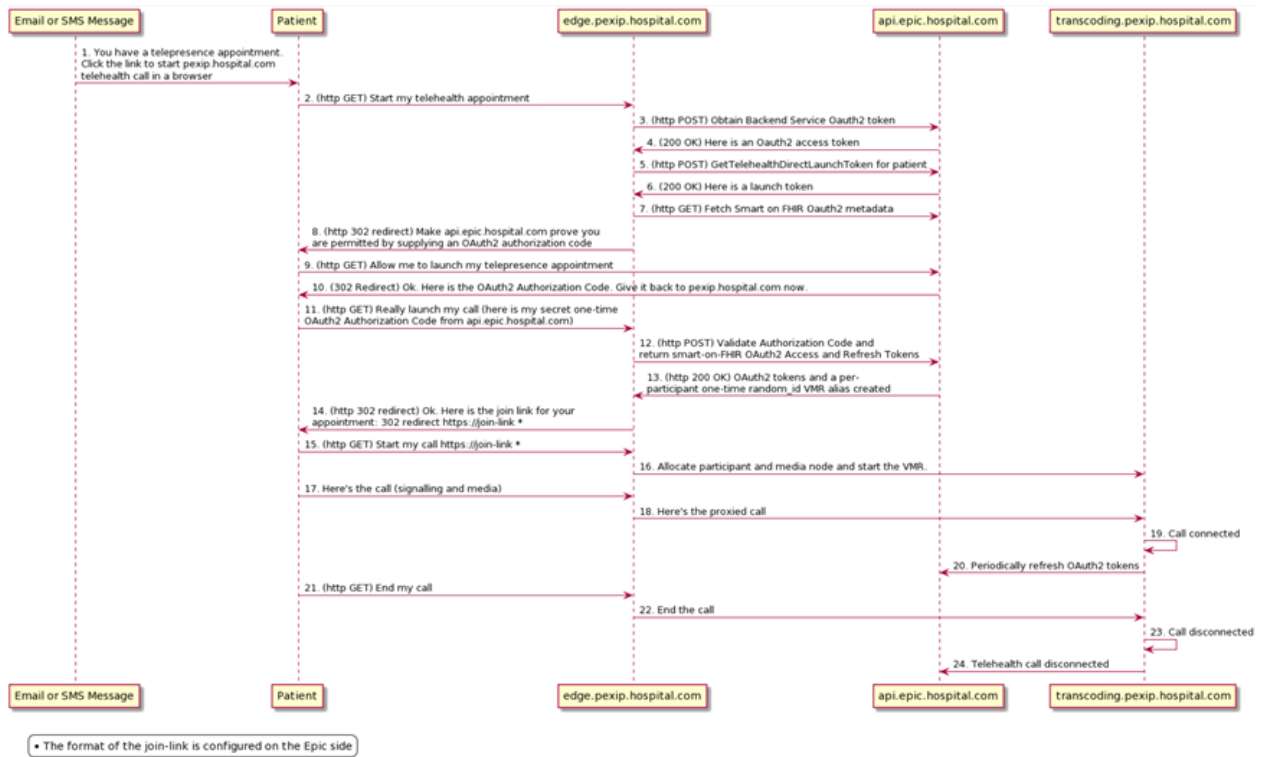
Pexip Infinity needs to access Epic at several points while setting up a telehealth call:

1. During OAuth authentication before the call starts (after a user has proven their identity to Epic's satisfaction they get redirected back to Pexip Infinity with a one-time-use ID which Pexip needs to take and exchange for Epic OAuth2 tokens).
2. When the call is connected and while it is ongoing, Pexip may need to periodically renew its OAuth2 tokens as they typically have a finite lifetime of approximately 1 hour.
3. When the call ends, Pexip uses its OAuth2 token to make an API call to let Epic know that the call has ended.

The following sequence diagram shows the call flow when establishing and finishing a telehealth call via the Epic apps. It shows the touch points between the Epic app, the patient/provider, the Pexip proxying and transcoding nodes and the API calls to the Epic server.



This diagram shows the flow when a patient joins via an SMS or email notification (step 7 onwards in this flow is the same as step 4 onwards in the Epic apps flow above):



Note that:

- The Epic API (at `api.epic.hospital.com` in our example) is publicly reachable.
- All transcoding nodes (and proxying nodes) need to be able to reach out to `api.epic.hospital.com`. There are no inbound messages to transcoding nodes.
- You can use a reverse proxy instead of a proxying node (the addresses are configured in the Epic telehealth profile).

Integrating Pexip Infinity with authentication and provisioning services

Pexip Infinity can be configured to connect to a Windows Active Directory LDAP server, or any other LDAP-accessible database, in order to authenticate and authorize the login accounts that are allowed to connect to the Pexip Infinity Administrator interface or the Pexip Infinity API, and to bulk-provision individual Virtual Meeting Rooms or devices for every member of the directory.

Pexip Infinity can integrate with Active Directory Federation Services (AD FS) to provide Infinity Connect clients and other third-party applications with single sign-on access. This allows users to register their clients using their AD credentials.

For information about how to deploy Pexip Infinity with these authentication and provisioning services, see:

Managing administrator access via LDAP

You can configure the Pexip Infinity platform to authenticate and authorize administrator login accounts via a centrally managed LDAP-accessible server. This provides increased security and more flexibility than using the single administrator account that is installed by default.

Instead of authenticating the supplied username and password credentials against its own internal database, Pexip Infinity contacts the LDAP server to authenticate the administrator's user account. It then uses the account's LDAP group attributes in combination with role mappings defined in Pexip Infinity to determine which Pexip Infinity features the administrator is authorized to access.

You can also configure the Pexip Infinity platform for client certificate authentication. This means that instead of logging in to the Pexip Infinity Administrator interface via the standard login page, or providing an authorization header when accessing the management API, administrators present (via their browser) a client certificate containing their user identification details. The validation of the presented certificate acts as the authentication phase and the username attributes in the certificate are used to determine which features the administrator is authorized to access.

The configuration described here applies to all administrator accounts connecting to the Pexip Infinity Administrator interface or the Pexip Infinity API. It does not apply to SSH connections. When using LDAP authentication, Pexip Infinity is configured by default to work with a Windows Active Directory LDAP server, but it can also be configured to work with other LDAP-accessible databases.

All usernames and passwords are case sensitive.

The following sections describe:

- [Configuration summary for LDAP authentication](#)
- [Configuring how administrators are authenticated](#)
- [Configuring administrator roles](#)
- [Configuring LDAP role mappings](#)
- [Examples: configuring permissions for an AD group](#)
- [Reinstating the local admin account](#)

Configuration summary for LDAP authentication

To enable authentication and authorization via LDAP, you need to configure both the LDAP database (if it is not already configured with user details) and the Pexip Infinity platform.

The LDAP database must be configured with:

- administrators' user credentials
- groups that define the capabilities of the users.

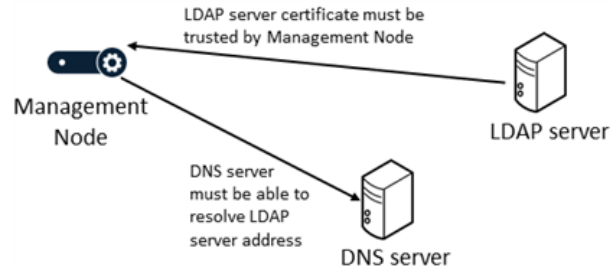
The Pexip Infinity platform must be configured with:

- an authentication source setting that uses an LDAP database
- connection details for the LDAP server; if the server address is an FQDN, ensure that it is resolvable over the DNS server configured for the Management Node
- administrator roles to control the actions that administrators can perform
- LDAP role mappings that map LDAP groups to administrator roles.

If a secure TLS connection between the LDAP server and the Management Node is required, ensure that:

- The LDAP server address is specified as an FQDN (so that it matches the name on the certificate presented by the LDAP server, which is typically created for the host name rather than the IP address).
- The Management Node trusts the certificate presented by the LDAP server; typically this means that the LDAP server certificate has to be uploaded to the Management Node as a trusted CA certificate (as the LDAP server's certificate is often generated by an internal authority which would not be included in Pexip's inbuilt list of trusted CA certificates).

Note that the Management Node's server certificate does not have to be trusted by the LDAP server (unless the LDAP server has been explicitly configured to demand a client certificate).




The Pexip Infinity platform configuration steps for specifying an LDAP authentication source, and configuring administrator and LDAP role mappings are described in more detail in the following sections, and there is an [example](#) that shows how to configure permissions for an AD group. For information about installing server and trusted CA certificates, see [Managing TLS and trusted CA certificates](#).

Configuring how administrators are authenticated

To configure how administrators are authenticated when they log in to the Pexip Infinity Administrator interface or API, go to **Users & Devices > Administrator Authentication**. The options are:

Option	Description
Authentication source	<p>The database to query for administrator authentication and authorization.</p> <p>Local database: uses the Pexip Infinity local on-box database. Administrators can only log in via the default account (typically admin) and will have full administrator privileges.</p> <p>LDAP database: administrators log in using an account configured on the LDAP database and obtain privileges according to the groups and roles associated with that account. Note that if this option is selected and the LDAP server is inaccessible for any reason, administrators will not be able to log in to the Pexip Infinity web-based Administrator interface or API.</p> <p>LDAP database and local database: administrators can log in using either the default local admin account or via an account configured on the LDAP database.</p> <p>When using an LDAP database, you must configure the items in the LDAP configuration section. By default, Pexip Infinity checks the entered username against the Active Directory <code>sAMAccountName</code> attribute (as configured in the LDAP user search filter advanced setting below).</p> <p>Default: <i>Local database</i>.</p>

Option	Description
Require client certificate	<p>Controls whether administrators are authenticated via a client certificate. By default, administrators log in to the Pexip Infinity Administrator interface via the standard login page, and provide an authorization header when accessing the management API. Instead, users can be required to present (via their browser) a client certificate containing their user identification details. The options are:</p> <p>Not required: Client certificates are not required. Administrators log in via the standard login page and provide a password which is authenticated against the selected Authentication source. Management API requests require an authorization header.</p> <p>Required (user identity in subject CN): administrators identify themselves via the identity contained in the subject CN (common name) of the client certificate presented by their browser.</p> <p>Required (user identity in subjectAltName userPrincipalName): administrators identify themselves via the identity contained in the subjectAltName userPrincipalName attribute of the client certificate presented by their browser.</p> <p>Default: Not required.</p> <p> When a client certificate is required, the standard login page is no longer presented. Administrators will not be able to access the Pexip Infinity Administrator interface or the management API if their browser does not present a valid certificate that contains a user identity which exists in the selected Authentication source.</p>
LDAP configuration	
LDAP server address	<p>The domain name (for DNS SRV lookup), FQDN (for DNS A/AAAA lookup) or IP address of the LDAP server. If using a domain or an FQDN, ensure that it is resolvable over DNS.</p> <p>You must also ensure that Pexip Infinity has trusted CA certificates for the authority that signed the LDAP server's certificate (if a TLS connection is required).</p> <p>We strongly recommend that you do not use an IP address. If an IP address is used, and a TLS connection is required, this will only work if the IP address is specified as the common name in the LDAP server's certificate.</p> <p>See Troubleshooting LDAP server connections for more information about how the system establishes a connection to the LDAP server and how to troubleshoot connection issues.</p>
Allow insecure transport	By default the system will attempt to establish a secure TLS connection with the LDAP server. Select this option if you want to allow the system to fall back to a TCP connection (using SASL DIGEST-MD5). You cannot specify the LDAP server by IP address if this option is selected.
LDAP bind username and password	The username and password of the bind account on the LDAP server. This should be a domain user service account, not the Administrator account.
LDAP base DN	The base DN (distinguished name) of the LDAP forest to query (e.g. <code>dc=example,dc=com</code>).
Advanced LDAP configuration	
By default the advanced LDAP configuration settings are preconfigured for Windows Active Directory, and may also be appropriate for other LDAP databases such as OpenLDAP.	
Search global catalog	Select this option to expand the scope of the search to the entire Active Directory Global Catalog instead of traditional LDAP. Note that this uses ports 3268 (TCP) and 3269 (TLS).
LDAP user search DN	The DN relative to the LDAP base DN to query for user records (e.g. <code>ou=people</code>). If blank, the LDAP base DN is used. In deployments with large user bases, you may want to configure this to optimize the LDAP user queries.
LDAP user filter	<p>The LDAP filter used to match user records in the directory.</p> <p>Default: <code>(&(objectclass=person)!((objectclass=computer)))</code></p>

Option	Description
LDAP user search filter	<p>The LDAP filter used to find user records when given the user name. The filter may contain {username} to indicate locations into which the username is substituted. This filter is applied in conjunction with the LDAP user filter and must contain at least one substitution.</p> <p>If client certificate-based authentication is used, this filter usually must include 'userPrincipalName={username}' either in addition to, or instead of, the default value; for example '((uid={username})(sAMAccountName={username})(userPrincipalName={username}))'.</p> <p>To log in using an email address, you can use '((uid={username})(sAMAccountName={username})(mail={username}))' — note that this requires the use of LDAPS.</p> <p>Default: ((uid={username})(sAMAccountName={username}))</p>
LDAP group attributes	<p>A comma-separated list of attributes in the LDAP user record to examine for group membership information. The attribute value must contain the DN of each group the user is a member of. If no attributes are specified, or none of the specified attributes are present in the LDAP user record, an LDAP group search (using the remaining advanced configuration options below) is performed instead.</p> <p>Default: memberOf</p>
LDAP group search DN	<p>The DN relative to the LDAP base DN to query for group records (e.g. ou=groups) when no group attributes are present in the LDAP user record. If blank, the LDAP base DN is used. In deployments with large user bases, you may want to configure this to optimize the LDAP group queries.</p>
LDAP group filter	<p>The LDAP filter used to match group records in the directory.</p> <p>Default: ((objectclass=group)(objectclass=groupOfNames)(objectclass=groupOfUniqueNames)(objectclass=posixGroup))</p>
LDAP group membership filter	<p>The LDAP filter used to search for group membership of a user. The filter may contain {userdn} to indicate locations into which the user DN is substituted. The filter may contain {userid} to indicate locations into which the user UID is substituted. This filter is applied in conjunction with the LDAP group filter and must contain at least one substitution.</p> <p>Default: ((member={userdn})(uniquemember={userdn})(memberuid={userid}))</p>

If authentication against an LDAP database is configured, you can save the settings only if Pexip Infinity can successfully contact the specified LDAP server.

Note that all LDAP distinguished names must be entered as per the LDAP standard ([RFC 4514](#)). LDAP configuration is case insensitive.

Supporting nested security groups in Windows Active Directory

The default LDAP configuration does not support nested security groups in Windows Active Directory. For example, if group A is allowed to log in via LDAP, and if group B is a member of group A, then any user who is only a member of group B will not be allowed to log in.

To allow members of a nested Active Directory security group to log in over LDAP:

1. Go to **Users & Devices > User Authentication** and expand the **Advanced LDAP configuration** section.
2. Ensure that **LDAP group attributes** is empty (i.e. remove the default "memberOf" content).
3. Change **LDAP group membership filter** to "(member:1.2.840.113556.1.4.1941:={userdn})"
4. Select **Save**.

(This configuration uses the LDAP_MATCHING_RULE_IN_CHAIN OID. More information on this can be found at <https://msdn.microsoft.com/en-us/library/aa746475%28VS.85%29.aspx>.)

Configuring administrator roles

Administrator roles control the actions that administrators can perform in the web-based Administrator interface or management API after they have been authenticated. For example, you can configure a role that allows an administrator to only view (and not modify) specific items of configuration data or status information via the Administrator interface.

When an administrator has restricted permissions, all navigation menu options are still displayed, but they are given an **Access denied** message if they try to select a menu option that they are not authorized to use. For read-only restrictions, the relevant **Add <item>** options are not displayed.

Two roles are present by default:

- **Read-only:** allows read-only access to all configuration settings and status information when accessing the system via the web-based Administrator interface or the API.
- **Read-write:** allows full administrative access when accessing the system via the web-based Administrator interface or the API.

To add, edit or delete administrator roles, go to **Users & Devices > Administrator Roles**. When configuring roles, the options are:

Option	Description
Name	A descriptive name of the role, e.g. "auditor" or "management system".
Permissions	<p>Select from the list of Available permissions the set of permitted actions for the role and then use the right arrow to move the selected actions into the Chosen permissions list. For more information on each permission, see Managing administrator access via LDAP.</p> <p>i All roles must include the <i>Is an administrator</i> permission for access to the system. In addition, the <i>May use web interface</i> and <i>May use API</i> permissions must be included for access via the web-based Administrator interface and API respectively. You must then also add all of the other permissions, such as <i>May modify system configuration</i> and so on, that you want to apply to the role — if a role has, for example, only the <i>Is an administrator</i> and <i>May use web interface</i> permissions, an administrator with that role will be able to log in via the web-based Administrator interface but will not be able to perform any actions.</p>

The permissions that are applied to the default *Read-only* role are shown below:

Change Administrator role

Name

Permissions

The permitted actions for accounts with this role

Available permissions

Filter

- May restore system backup
- May add/remove gateways
- May add/remove themes
- May add/remove VMRs
- May download TLS private key
- May modify authentication configuration
- May modify conference status
- May modify gateway configuration
- May configure logs
- May create/delete packet capture
- May create/delete system backup
- May modify system configuration
- May modify VMR configuration

Choose all

Chosen permissions

- Is an administrator
- May use API
- May use web interface
- May view authentication configuration
- May view conference status
- May view gateway configuration
- May view logs
- May download packet capture
- May download system backup
- May view system configuration
- May generate system snapshot
- May view system status
- May view VMR configuration


Remove all

Configuring LDAP role mappings

LDAP role mappings are used to map the LDAP groups associated with LDAP user records to the Pexip Infinity administrator roles. You must configure a separate LDAP role mapping for each LDAP group for which you want to map one or more Pexip Infinity administrator roles.

To add, edit or delete LDAP role mappings, go to **Users & Devices > LDAP Role Mappings**. When configuring LDAP role mappings, the options are:

Option	Description
Name	A descriptive name of the role mapping, e.g. "domain administrator with full privileges".

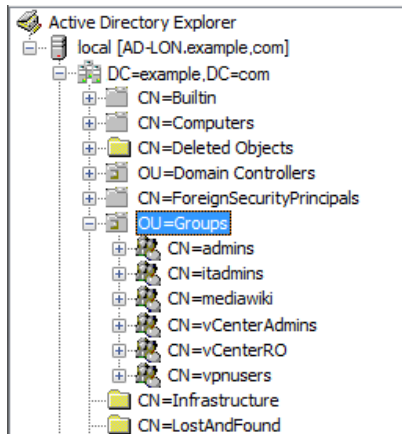
Option	Description
LDAP group DN	<p>Select the LDAP group against which you want to map one or more administrator roles.</p> <p>The list of LDAP groups is only populated when there is an active connection to an LDAP server (Users & Devices > Administrator Authentication).</p> <p>Note that the LDAP groups used for role mappings cannot be the pre-defined AD groups such as Domain Users etc. but need to be explicitly configured custom groups.</p>
Roles	<p>Select from the list of Available roles the administrator roles to associate with the LDAP group and then use the right arrow to move the selected roles into the Chosen Roles list.</p> <p>All of the underlying permissions within a role are "positive" permissions, i.e. they allow the administrator to do something. If more than one role is selected, all of the permissions associated with each role are combined and granted to the relevant administrator.</p> <p>Note that you can select  which opens a new window from where you can configure a new administrator role. When you save the role it is automatically added to the set of Chosen Roles.</p>

Examples: configuring permissions for an AD group

These examples show how you can configure the specific actions (permissions) that all members of an AD group are allowed to perform when administering Pexip Infinity, and provide methods to filter the groups that are displayed.

The filtering options are not mandatory but they do make it easier to select the appropriate LDAP groups, and can optimize system performance.

Let's assume that you have the following set of groups already configured in Windows Active Directory:



In both of the examples below you need to ensure that you have configured an LDAP authentication source (**Users & Devices > Administrator Authentication**) that can access your AD server, for example:

Change Administrator authentication

Authentication source	LDAP database ▼ *	The database to query for administrator authentication and authorization.
Require client certificate	Not required ▼ *	Whether to require a client TLS certificate for user authentication.
LDAP configuration		
LDAP server address	ldap.server	The hostname the LDAP server. Maximum length: 255 characters.
Allow insecure transport	<input type="checkbox"/>	Permit LDAP queries to be sent over an insecure connection.
LDAP bind username	admin	The username used to bind to the LDAP server. This should be a domain user service account, not the Administrator account. Maximum length: 255 characters.
LDAP bind password	*****	The password used to bind to the LDAP server. Maximum length: 100 characters.
LDAP base DN	dc=example,dc=com	The base DN of the LDAP forest to query (e.g. dc=example,dc=com). Maximum length: 255 characters.

Example: Filtering by specifying an LDAP group search DN

This example shows how to configure all AD users who are members of the "itadmins" group to be able to add, modify and delete VMR/conference related settings, but only be able to view other configuration aspects of Pexip Infinity (system settings, logs etc).

To make it easier to select the "itadmins" group we have specified an **LDAP group search DN** to limit the number of LDAP groups that are presented when configuring your LDAP roles:

1. Go to **Users & Devices > Administrator Authentication** where your LDAP configuration has been completed, as shown above, and open the **Advanced LDAP Configuration** section.
2. In this case, we want to define permissions based upon membership of specific AD groups, therefore we have configured the **LDAP group search DN** setting to **ou=groups**.

This means that when we configure the LDAP roles, the set of LDAP groups that is presented is filtered to include only those in the groups organizational unit (ou).

Advanced LDAP configuration (Hide)

Search global catalog ☐ Search the Active Directory Global Catalog instead of traditional LDAP.

LDAP user search DN The DN relative to the base DN to query for user records (e.g. ou=people). If blank, the base DN will be used. Maximum length: 255 characters.

LDAP user filter * The LDAP filter used to match user records in the directory. Default: (&(objectclass=person)(!(objectclass=computer))). Maximum length: 255 characters.

LDAP user search filter * The LDAP filter used to find user records when given the user name. The filter may contain {username} to indicate locations into which the username is substituted. This filter will be applied in conjunction with the LDAP user filter and must contain at least one substitution. Default: ((uid={username})(sAMAccountName={username})). Maximum length: 255 characters.

LDAP group attributes A comma-separated list of attributes in the LDAP user record to examine for group membership information. The attribute value must contain the DN of each group the user is a member of. If no attributes are specified, or none of the specified attributes are present in the LDAP user record, an LDAP group search will be performed, instead. Default: memberOf. Maximum length: 100 characters.

LDAP group search DN The DN relative to the base DN to query for group records (e.g. ou=groups). If blank, the base DN will be used. Maximum length: 255 characters.

LDAP group filter * The LDAP filter used to match group records in the directory. Default: ((objectclass=group)(objectclass=groupOfNames)(objectclass=groupOfUniqueNames)(objectclass=posixGroup)). Maximum length: 255 characters.

LDAP group membership filter * The LDAP filter used to search for group membership of a user. The filter may contain {userdn} to indicate locations into which the user DN is substituted. The filter may contain {userid} to indicate locations into which the user UID is substituted. This filter will be applied in conjunction with the LDAP group filter and must contain at least one substitution. Default: ((member={userdn})(uniquemember={userid})). Maximum length: 255 characters.

- We now need to configure an administrator role (**Users & Devices > Administrator Roles**) that defines the set of actions that can be performed by administrators who have been assigned that role.

Here, a "Manage Conferences" role has been created. The Chosen permissions allow an administrator to use the web interface to configure all service-related items such as VMRs, themes, gateway rules, but to only be able to view (and not modify) all other configuration.

Add Administrator role

Name *

The permitted actions for accounts with this role

Permissions

Available permissions

- May modify authentication configuration
- May configure logs
- May modify system configuration
- May use API
- May generate system snapshot

Choose

Chosen permissions

- May add/remove gateways
- May add/remove themes
- May add/remove VMRs
- Is an administrator
- May modify conference status
- May modify gateway configuration
- May modify VMR configuration
- May use web interface
- May view authentication configuration
- May view conference status
- May view gateway configuration
- May view logs
- May view system configuration
- May view system status
- May view VMR configuration

Remove all

- The final step is to associate this administrator role with an LDAP role/group (**Users & Devices > LDAP Role Mappings**).

Here, we have configured an "IT admins - manage conferences" role. The **LDAP group DN** drop-down presents a list of LDAP groups from AD. In our case this list is filtered to only show those groups in the **ou=groups** organizational unit (due to the LDAP group search DN configuration in step 2).

We have selected the **itadmins** group and associated it with the **Manage Conferences** role we created in step 3. (Note that you can associate the LDAP role with more than one administrator role if required.)

Add LDAP role mapping

Name *

The name of the role mapping. Maximum length: 250 characters.

LDAP group DN *

Maximum length: 255 characters.

Roles

Read-write
Read-only

Chosen Roles

Manage Conferences

[Choose all](#) [Remove all](#)

The role(s) to assign matching users to. Hold down "Control", or "Command" on a Mac, to select more than one.

[Save](#) [Save and add another](#)

This means that AD users who are in the itadmins group can now sign in to the Pexip Infinity Administrator interface, using their AD credentials, and configure service-related settings only.

To set up different permissions for members of other AD groups, repeat steps 3 and 4 to create different administrator role and LDAP role associations.

Example: Limiting groups by specifying an LDAP group filter

This example is similar to the example above, but shows an alternative method of limiting the number of LDAP groups that are presented when configuring your LDAP roles.

In this case we show how to specify an LDAP group filter to limit the groups that are displayed.

1. Go to **Users & Devices > Administrator Authentication** where your LDAP configuration has been completed, as shown above, and open the **Advanced LDAP Configuration** section.
2. In this example, we are specifying a group filter so that when we configure the LDAP roles, the set of LDAP groups that is presented is filtered to only show those whose name starts with "vc".

We do this by configuring the LDAP group filter to `(&(objectclass=group)(cn=vc*))`

Advanced LDAP configuration (Hide)

Search global catalog ☐
Search the Active Directory Global Catalog instead of traditional LDAP.

LDAP user search DN
The DN relative to the base DN to query for user records (e.g. ou=people). If blank, the base DN will be used. Maximum length: 255 characters.

LDAP user filter *
The LDAP filter used to match user records in the directory. Default: (&(objectclass=person)(!(objectclass=computer))). Maximum length: 255 characters.

LDAP user search filter *
The LDAP filter used to find user records when given the user name. The filter may contain {username} to indicate locations into which the username is substituted. This filter will be applied in conjunction with the LDAP user filter and must contain at least one substitution. Default: ((uid={username})(sAMAccountName={username})). Maximum length: 255 characters.

LDAP group attributes
A comma-separated list of attributes in the LDAP user record to examine for group membership information. The attribute value must contain the DN of each group the user is a member of. If no attributes are specified, or none of the specified attributes are present in the LDAP user record, an LDAP group search will be performed. Instead. Default: memberOf. Maximum length: 100 characters.

LDAP group search DN
The DN relative to the base DN to query for group records (e.g. ou=groups). If blank, the base DN will be used. Maximum length: 255 characters.

LDAP group filter *
The LDAP filter used to match group records in the directory. Default: ((objectclass=group)(objectclass=groupOfNames)(objectclass=groupOfUniqueNames)(objectclass=posixGroup)). Maximum length: 255 characters.

LDAP group membership filter *
The LDAP filter used to search for group membership of a user. The filter may contain {userdn} to indicate locations into which the user DN is substituted. The filter may contain {userid} to indicate locations into which the user UID is substituted. This filter will be applied in conjunction with the LDAP group filter and must contain at least one substitution. Default: ((member={userdn})(uniquemember={userid})). Maximum length: 255 characters.

- As with the previous example, you need to ensure that you have configured an administrator role (Users & Devices > Administrator Roles), such as "Manage Conferences", that defines the set of actions that can be performed by administrators who have been assigned that role.
- The final step is to associate this administrator role with an LDAP role/group (Users & Devices > LDAP Role Mappings). Here, we have configured a "vCenter Admins - Manage Conferences" role. The LDAP group DN drop-down presents a list of LDAP groups from AD. In our case this list is filtered to only show those groups whose name starts with "vc".

Add LDAP role mapping

Name *
The name of the role mapping. Maximum length: 250 characters.

LDAP group DN *
vCenterAdmins
vCenterAdmins
vCenterRO
to match. Maximum length: 255 characters.

Roles

Available Roles

Filter

Read-write
Read-only

Choose all

Chosen Roles

Manage Conferences

Remove all

The role(s) to assign matching users to. Hold down "Control", or "Command" on a Mac, to select more than one.

This means that AD users who are in the vCenterAdmins group can now sign in to the Pexip Infinity Administrator interface.

Reinstating the local admin account

If necessary you can reinstate access via the Pexip Infinity local on-box database, so that administrators can log in via the default account (typically **admin**) and will have full administrator privileges. You may need to do this if, for example, the **Authentication source** is configured as **LDAP database** and your connectivity to the LDAP server goes down or your credentials become invalid.

To reactivate your local admin account:

1. Log in to the Management Node over SSH.
2. For local admin access only, run the command:

```
authset LDAP LOCAL
```


or, for LDAP and local admin access, run the command:

```
authset LDAP BOTH
```

You can also disable client certificate authentication so that you can log in to the Pexip Infinity Administrator interface via the standard login page.

To disable certificate-based authentication:

1. Log in to the Management Node over SSH.
2. Run the command:

```
authset CBA OFF
```

If you forget the password for the Pexip Infinity Administrator interface, you can [re-run the installation wizard](#), being sure to change only the **Web administration password** setting.

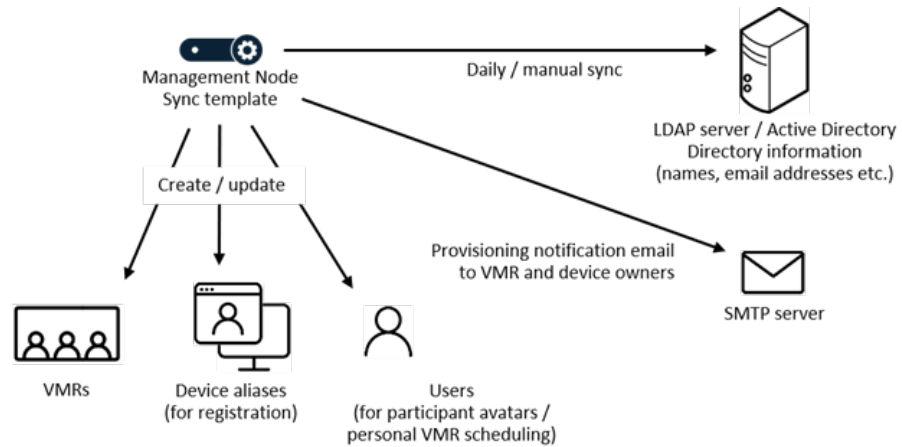
Provisioning VMRs, devices and users from Active Directory via LDAP

In large organizations with many employees and users of video conferencing, you may need to configure lots of Virtual Meeting Rooms (VMRs) and associated records to support those employees. Much of this data can be bulk-provisioned from directory information contained in a Windows Active Directory LDAP server, or any other LDAP-accessible database.

The following Pexip Infinity configuration settings can be provisioned directly from an LDAP data source:

- **Virtual Meeting Rooms (VMRs)**: this provides a simple way to automatically provide a personal VMR for every employee in an organization. Data such as employee names can be imported from the directory and used to generate a unique name and alias for each VMR, following a pattern such as **meet.<username>@example.com**. Other VMR settings such as PIN numbers, the layout and theme can also be assigned, depending upon your VMR template configuration.
- **Device aliases**: these are the aliases that people can use to register their endpoint or Infinity Connect client to a Conferencing Node, and how those devices can be authenticated.
- **User records**: these are required for two optional features:
 - Scheduling meetings in personal VMRs — part of the VMR Scheduling for Exchange feature. See [Enabling VMR scheduling in personal VMRs](#) for more information.
 - Participant avatars — conference participants and directory contacts within Pexip Infinity can be represented by an avatar or image. You can configure user records to represent those participants/contacts and associate each user with an avatar URL that points to an external service (such as Gravatar) which can be used to retrieve that user's avatar/image. The user's Email address is used as the primary identifier of each user record, and this must match the **Owner's email address** associated with the device alias of the conference participant. To use your avatar URLs, you must set up a policy profile that has **Use local avatar configuration** enabled. See [Applying user records](#) for more information.

You control which data is provisioned by setting up one or more **sync templates** in Pexip Infinity. Each sync template can be configured to automatically synchronize against its LDAP **sync source** once per day. This ensures that the Pexip Infinity VMR, device and user configuration stays in step with all additions and removals that have occurred in the source directory. Many of the VMR/device/user settings can be tagged as "overridable" which means that if that setting (the VMR PIN number for example) is manually overridden after the VMR, device or user record has been created, a subsequent resynchronization will not reset it back to another value.



If Pexip Infinity is configured with the details of an SMTP server, you can send an email to the VMR owner whenever a synchronization creates a new VMR or modifies an existing VMR, or to inform a user of the device alias and credentials that they need to use to register their device. Emails are not sent for any user records that are created.

Configuration summary

To provision your Virtual Meeting Rooms, device aliases or user records from an LDAP database you must:

1. [Configure the connection details](#) of an LDAP data source. You must also ensure that Pexip Infinity has trusted CA certificates for the authority that signed the LDAP server's certificate (if a TLS connection is required).
2. [Configure an LDAP synchronization template](#) that controls what to synchronize — VMRs, device aliases or users — and how individual settings such as the VMR name, dialable alias and PIN numbers are generated, which of those values can be manually [overridden](#), and whether to synchronize automatically against the LDAP data source.
If required, you can configure multiple synchronization templates that use the same LDAP data source (or different templates that use different LDAP sources). For example, you may want to use a different template for VMR syncing than the template used for device alias syncing if you need to use different LDAP filter criteria for those two user bases.
3. [Configure an SMTP server and construct an email template](#), if you want to send a provisioning notification email to the VMR owner or intended user of a device alias.
4. [Generate the VMRs, device aliases and user records](#) using your template and LDAP data source. Set the template to sync automatically when you are happy that the template is configured correctly.

These configuration steps are described in more detail in the following sections.

In addition, you can also:

- [View the status](#) of ongoing or completed LDAP template synchronization processes via **Status > LDAP Sync**.
- [Delete](#) all of the VMRs, devices and users that have been created from a specific template.

Configuring an LDAP data source

To configure Pexip Infinity with the connection details of an LDAP data source, such as a Windows Active Directory LDAP server:

1. Go to **Utilities > LDAP Sync Sources**.
2. Select **Add LDAP sync source**, and then complete the following fields:

Option	Description
Name	The name used to identify this LDAP sync source.
Description	An optional description of the sync source.

Option	Description
LDAP server address	<p>The domain name (for DNS SRV lookup), FQDN (for DNS A/AAAA lookup) or IP address of the LDAP server. If using a domain or an FQDN, ensure that it is resolvable over DNS.</p> <p>You must also ensure that Pexip Infinity has trusted CA certificates for the authority that signed the LDAP server's certificate (if a TLS connection is required).</p> <p>We strongly recommend that you do not use an IP address. If an IP address is used, and a TLS connection is required, this will only work if the IP address is specified as the common name in the LDAP server's certificate.</p> <p>See Troubleshooting LDAP server connections for more information about how the system establishes a connection to the LDAP server and how to troubleshoot connection issues.</p>
Search global catalog	Select this option to expand the scope of the search to the entire Active Directory Global Catalog instead of traditional LDAP. Note that this uses ports 3268 (TCP) and 3269 (TLS).
Allow insecure transport	By default the system will attempt to establish a secure TLS connection with the LDAP server. Select this option if you want to allow the system to fall back to a TCP connection (using SASL DIGEST-MD5). You cannot specify the LDAP server by IP address if this option is selected.
LDAP bind username and password	The username and password of the bind account on the LDAP server. This should be a domain user service account, not the Administrator account.
LDAP base DN	The base DN (distinguished name) of the LDAP forest to query (e.g. <code>dc=example,dc=com</code>).

3. Select Save.

You can save the sync source details only if Pexip Infinity can successfully contact the specified LDAP server.

Note that all LDAP distinguished names must be entered as per the LDAP standard ([RFC 4514](#)). LDAP configuration is case insensitive.


Configuring an LDAP synchronization template

LDAP synchronization templates control what to synchronize — VMRs, device aliases and/or users — and how individual settings such as the VMR name, dialable alias and PIN numbers are generated, and what to include in provisioning notification emails sent to the VMR or device owner. You must also tell the template which LDAP data source to use.

Many of the VMR, device and user settings are based on **patterns** in the template that define how that setting is generated (such as using an email address as the device alias). See [Using templates, variables and filters when provisioning VMRs, devices and users](#) for more information.

To configure Pexip Infinity with an LDAP synchronization template:

1. Go to **Utilities > LDAP Sync Templates**.
2. Select **Add LDAP sync template**, and then complete the following fields:

Option	Description
Name	The name used to identify this synchronization template.
Description	An optional description of the synchronization template.
LDAP user search DN	The DN relative to the LDAP base DN of the sync source to query for user records (e.g. <code>ou=people</code>). If blank, the LDAP base DN is used. In deployments with large user bases, you may want to configure this to optimize the LDAP user queries.
LDAP user filter	<p>The LDAP filter used to match user records in the directory.</p> <p>Default: <code>(&(objectclass=person)!(objectclass=computer))</code></p> <p>For more information, see LDAP search and filter examples.</p>
LDAP sync source	<p>Select the LDAP data source to use when syncing records.</p> <p>Note that you can select  which will open a new window from where you can configure a new sync source.</p>

Option	Description
Sync VMRs	Enables VMR synchronization for this template. Default: enabled.
Sync devices	Enables device alias synchronization for this template. Default: disabled.
Sync users	Enables user record synchronization for this template. Default: disabled.
Enable automatic daily sync	<p>Select this option to instruct Pexip Infinity to automatically synchronize this template against its LDAP sync source once per day. This ensures that VMRs, devices and users are regularly updated, deleted or created as appropriate based on the latest data in the sync source. Therefore, for example, if a user is removed from Active Directory or their account is disabled, their corresponding VMR, device or user record in Pexip Infinity will be deleted via the next daily sync.</p> <p>As template synchronization can result in the automatic creation, modification or deletion of large numbers of VMRs, devices and users, we recommend that you only enable automatic syncing after you have manually synced at least once and have verified that you are satisfied with your sync template configuration.</p> <p>All automatic synchronizations are initiated at 01:00 UTC (this start time cannot be configured). After an initial sync, which can take several minutes in a large organization, an ongoing daily sync is typically much faster as it only processes changes since the previous sync.</p> <p>Default: disabled.</p>
Device settings (shown when Sync devices is selected)	
Device alias pattern	<p>The pattern for the alias that will be registered by the device and be used by people trying to call the device.</p> <p>For more information, see Using templates, variables and filters when provisioning VMRs, devices and users.</p> <p>Default: {{mail}}</p> <p>If a device with the generated alias already exists, that existing device configuration is left unchanged.</p>
Device tag pattern	The pattern for the unique identifier used to track usage of the device.
Allow device tag to be manually overridden	You can also allow the auto-generated device tag to be manually overridden for each device alias.
Device description pattern	The pattern to use when generating the optional description of this device alias.
Allow device description to be manually overridden	You can also allow the auto-generated device description to be manually overridden for each device alias.
Device username pattern	The pattern for the device username. Note that you can use the same username for different device aliases (although we recommend you only do this for multiple devices used by the same person).
Allow username to be manually overridden	<p>Default: {{mail}}</p> <p>You can also allow the auto-generated username to be manually overridden for each device alias.</p>
Device password pattern	The pattern to use when generating the password for this device. A password pattern must be specified if a username pattern is configured.
Allow password to be manually overridden	<p>Example: {{ ("some random string"+(mail pex_reverse)) pex_hash pex_tail(16) }}</p> <p>You can also allow the auto-generated password to be manually overridden (by the administrator) for each device alias.</p>

Option	Description
Sync disabled accounts	<p>Syncs all device aliases, even if the corresponding LDAP account is disabled.</p> <p>By default, device aliases are only provisioned if the corresponding LDAP account is enabled in the LDAP directory. If Sync disabled accounts is selected, a device alias will be created for LDAP accounts that are marked as disabled. This may be useful, for example, if you have a disabled machine account in LDAP corresponding to a SIP or H.323 room system — however it is not generally useful for staff accounts because if an employee leaves an organization you usually want their device record to be deleted automatically after their account is disabled in the corporate LDAP directory.</p> <p>Default: disabled.</p>
Device registration settings (shown when Sync devices is selected)	
Enable SIP registration	Allows this device alias to register over the SIP protocol.
Enable H.323 registration	Allows this device alias to register over the H.323 protocol.
Enable Infinity Connect registration (non-SSO)	Allows an Infinity Connect client to register using this alias (not using SSO services).
Enable Infinity Connect registration using SSO	Allows an Infinity Connect client to register using this alias, using Single Sign-On (SSO) services such as AD FS to authenticate the registration.
Allow registration settings to be manually overridden	Allows all of the auto-generated device registration settings to be manually overridden for each device.
VMR settings (shown when Sync VMRs is selected)	
VMR name pattern	<p>The pattern to use to generate the name of the VMR. You should structure this pattern to generate a unique VMR name.</p> <p>For more information, see Using templates, variables and filters when provisioning VMRs, devices and users.</p> <p>Example: <code>{{givenName}} {{sn}} VMR</code></p> <p>If a VMR with the generated name already exists, that existing VMR configuration is left unchanged.</p>
VMR description pattern	<p>The pattern to use to generate the VMR description. This field is optional.</p> <p>Examples: <code>{{givenName}} {{sn}}'s meeting room</code></p>
Allow VMR description to be manually overridden	<p>You could use a more advanced pattern such as:</p> <p><code>{{givenName}} {{sn}}'s {%if department %} {{department}}{%endif %} meeting room</code></p> <p>which will insert "<department name>" before "meeting room", but only if a value exists in the LDAP department field.</p> <p>You can also allow the auto-generated VMR description to be manually overridden for each VMR.</p>

Option	Description
Host PIN pattern	<p>The pattern to use to generate the Host PIN (if required).</p> <p>For example, if you want a VMR to keep the same random PIN after every resync you could use a pattern like this:</p> <pre>{% set pl=6 %}{% set hp=(("an ungu3ssable Pa\$\$phr4s3" ~mail) pex_hash pex_tail(pl)) %}{hp}}</pre> <p>where you:</p> <ul style="list-style-type: none"> set pl to the number of digits required in the PIN (it is set to 6 in this example, but can be from 4-20), and replace "an ungu3ssable Pa\$\$phr4s3" with your own passphrase string (we recommend at least 14 characters long). <p>This will generate a random number that is pl digits long, but it will always generate the same number per VMR (assuming the VMR owner's LDAP mail field remains unchanged). If you want to do a periodic update of all of the PINs associated with VMRs generated from this template, then just change the passphrase string and they will be updated to a new random PIN (assuming Allow PIN settings to be manually overridden is not selected).</p> <p>Note that the ~ operator is similar to a + and converts arguments, nulls etc. safely to strings.</p>
Allow Guests	<p>Yes: the conference can have two types of participants: Hosts and Guests. You must configure a Host PIN to be used by the Hosts. You can optionally configure a Guest PIN; if you do not configure a Guest PIN, Guests can join without a PIN, but the meeting will not start until the first Host has joined.</p> <p>No: all participants have Host privileges.</p> <p>Default: <i>No</i>.</p>
Guest PIN pattern	<p>The pattern to use to generate the Guest PIN (if required).</p> <p>Note that if you set a Host PIN and a Guest PIN for a VMR, those two PINs must be different. Therefore we recommend that you generate Host and Guest PINs of different lengths, or ensure that some aspect of each PIN will be different, for example:</p> <pre>{% set pl=6 %}{% set hp=(("an ungu3ssable Pa\$\$phr4s3" ~mail) pex_hash pex_tail(pl)) %}{% set gp=(("a different Pa\$\$phr4s3" ~mail) pex_hash pex_tail(pl)) %}{% if hp == gp %}{("00000000000000000000"~((gp int)+123)) pex_tail(pl)}{% else %}{gp}{% endif %}</pre> <p>This works in conjunction with the example above that is used to generate the Host PIN. The pattern includes the recipe used to generate the Host PIN (hp) and uses the same technique to generate a random Guest PIN (gp). It then checks if the Guest PIN accidentally clashes with the Host PIN. If it does clash, the Guest PIN is adjusted (it is incremented by 123 in this example).</p> <p>As with the Host PIN pattern, you should:</p> <ul style="list-style-type: none"> set pl to the number of digits required in the Host and Guest PINs, and replace "an ungu3ssable Pa\$\$phr4s3" and "a different Pa\$\$phr4s3" with your own passphrase strings (both strings must be different).
Allow PIN settings to be manually overridden	<p>Allows the auto-generated Host and Guest PIN settings to be manually overridden for each VMR.</p> <p>Note that this would also preserve the generated PIN value, even if it does not actually get manually overridden.</p>

Option	Description
Layout	The layout controls the maximum number of other participants that each participant will see, and how the participants are arranged on the screen. For more information, see Selecting the layout seen by participants .
Show names of participants	Default: <i>Large main speaker and up to 7 other participants (1+7 layout)</i> .
Show name of active speaker	If participant name overlays are enabled, the display names or aliases of all participants are shown in a text overlay along the bottom of their video image.
Allow layout settings to be manually overridden	You can also allow the type of layout and whether to show names of participants to be manually overridden for each VMR.
Theme	The theme for use with this VMR. For more information, see Customizing conference images and voice prompts using themes .
Allow theme to be manually overridden	Default: <i><use Default theme></i> (the global default theme is used). You can also allow the auto-generated theme to be manually overridden for each VMR.
VMR alias 1 pattern	The pattern to use to generate the alias (string) that participants can dial to join this VMR. You should structure this pattern to generate a unique alias for the VMR. Example: <code>meet.{{givenName lower}}.{{sn lower}}@example.com</code> or <code>meet.{{mail}}</code> If the generated alias is already assigned to another VMR, that alias is left assigned to that existing VMR.
Alias 1 description pattern	The pattern to use to generate a description of the alias. This field is optional. Example: <code>The alphanumeric URI alias</code>
VMR alias 2...8 pattern and descriptions	You can optionally enter patterns for a further 7 alternative aliases, such as an E.164 alias, to associate with this VMR. (Aliases 5-8 are configured by expanding the Advanced VMR alias options section.) Do not use the <code>pex_random_pin()</code> filter to generate numeric aliases as this is likely to generate duplicate aliases. Where possible, it is best to use a phone number, employee ID or other typically unique value as the basis of a numeric alias. If a unique value does not exist, a good fallback is to use a <code>pex_hash()</code> expression such as the following to generate a consistent 6-digit random E164-type alias from a hash of each user's email address: <code>{{ ("my alias passphrase"+(mail pex_reverse)) pex_hash pex_tail(6) }}</code>
Allow aliases to be overridden	(This option is configured by expanding the Advanced VMR alias options section.) Allows aliases and alias descriptions for a VMR to be added, removed or modified in any way. When enabled this means that after the initial creation of a VMR and its aliases, subsequent syncs will not change any of the aliases or their descriptions (including any which were created by VMR alias 1..8 pattern and Description 1..8 pattern even if those patterns are modified in this template).
Advanced options (shown when Sync VMRs is selected)	
Guests can present	Controls whether the Guests in the conference are allowed to present content.
Allow Guests can present to be manually overridden	<ul style="list-style-type: none"> <i>Yes</i>: Guests and Hosts can present into the conference <i>No</i>: only Hosts can present Default: <i>Yes</i> You can also allow the auto-generated Guests can present setting to be manually overridden for each VMR.
Enable chat	Whether chat messaging is enabled for the conference:
Allow enable chat to be manually overridden	<ul style="list-style-type: none"> <i>Use global chat setting</i>: as per the global configuration setting. <i>Yes</i>: chat is enabled. <i>No</i>: chat is disabled. Default: <i>Use global chat setting</i> . You can also allow the auto-generated enable chat setting to be manually overridden for each VMR.

Option	Description
Maximum inbound call bandwidth (kbps)	Specifying a maximum inbound call bandwidth will limit the bandwidth of media received by Pexip Infinity from each individual participant dialed in to this VMR. For more information see Managing and restricting call bandwidth .
Maximum outbound call bandwidth (kbps)	Specifying a maximum outbound call bandwidth will limit the bandwidth of media sent from Pexip Infinity to each individual participant dialed in to this VMR. For more information see Managing and restricting call bandwidth .
Allow maximum bandwidth settings to be manually overridden	You can also allow the auto-generated maximum inbound and outbound call bandwidth to be manually overridden for each VMR.
Maximum call quality	Controls the maximum call quality for participants connecting to this service:
Allow maximum call quality to be manually overridden	<ul style="list-style-type: none"> <i>Use global setting</i>: use the global maximum call quality setting. <i>SD</i>: each participant is limited to SD quality. <i>HD</i>: each participant is limited to HD (720p) quality. <i>Full HD (1080p)</i>: allows any endpoint capable of Full HD to send and receive its main video at 1080p. Default: <i>Use global setting</i>
Participant limit	This optional field allows you to limit the number of participants allowed to join this VMR. For more information see Limiting the number of participants .
Allow participant limit to be manually overridden	You can also allow the auto-generated participant limit to be manually overridden for each VMR.
Service tag pattern	This optional field lets you assign a unique identifier to this service, which you can then use to track use of the service .
Allow service tag to be manually overridden	You can also allow the auto-generated service tag to be manually overridden for each VMR.
Conference capabilities	Allows you to limit the media content of the conference. For more information, see Controlling media capability .
Allow conference capabilities to be manually overridden	Default: <i>Main video + presentation</i> . You can also allow the auto-generated conference capabilities to be manually overridden for each VMR.
Media encryption	Controls the media encryption requirements for participants connecting to this service.
Allow media encryption to be manually overridden	<ul style="list-style-type: none"> <i>Use global setting</i>: use the global media encryption setting. <i>Best effort</i>: each participant will use media encryption if their device supports it, otherwise the connection will be unencrypted. <i>Required</i>: all participants (including RTMP participants) must use media encryption. <i>No encryption</i>: all H.323, SIP and MS-SIP participants must use unencrypted media. (RTMP participants will use encryption if their device supports it, otherwise the connection will be unencrypted.) Default: <i>Use global setting</i> You can also allow the auto-generated media encryption setting to be manually overridden for each VMR.
User options (shown when Sync Users is selected)	

Option	Description
User description pattern	The pattern to use to generate the user description. This field is optional. Examples: The user for <code>{{displayName}}</code>
Allow user description to be manually overridden	You could use a more advanced pattern such as: The user for <code>{{displayName}}</code> <code>{% if title or department%}{% if title %}{title}, {%endif%} {%if department %}{department}{% endif %}{% endif %}</code> which will include the user's job title and department name if they are specified in the LDAP source. You can also allow the auto-generated user description to be manually overridden for each user.
First name pattern	The pattern to use to generate the first name of the user. Default: <code>{% if givenName %}{givenName}{% endif %}</code>
Last name pattern	The pattern to use to generate the last name of the user. Default: <code>{% if sn %}{sn}{% endif %}</code>
Display name pattern	The pattern to use to generate the display name of the user. Example: <code>{% if displayName %}{displayName}{% endif %}</code> Note that the display name is not currently used in Pexip Infinity (for example, it does not affect participant name overlays and cannot be used when provisioning Infinity Connect clients).
Allow user names to be manually overridden	Allows the auto-generated user names to be manually overridden for each user.
Telephone number pattern	The pattern to use to generate the telephone number of the user. Default: <code>{{telephoneNumber pex_clean_phone_number}}</code>
Mobile number pattern	The pattern to use to generate the mobile number of the user. Default: <code>{{mobile pex_clean_phone_number}}</code>
Allow user contacts to be manually overridden	Allows the auto-generated user contact information (telephone and mobile numbers) to be manually overridden for each user.
Title pattern	The pattern to use to generate the title of the user. Default: <code>{% if title %}{title}{% endif %}</code>
Department pattern	The pattern to use to generate the department of the user. Default: <code>{% if department %}{department}{% endif %}</code>
Avatar URL pattern	The pattern to use to generate the avatar URL of the user. Note that: <ul style="list-style-type: none"> The avatar URL must be an unprotected resource (username and password credentials cannot be supplied with the request), and it must be reachable from Conferencing Nodes. The image retrieved from the avatar URL must be a JPEG. When a Conferencing Node sends an image request to the avatar URL, Pexip Infinity adds on extra URL parameters that specify the required dimensions, for example <code>?width=100&height=100&s=100</code> (the <code>s</code> is a size parameter used by Gravatar). Pexip Infinity only ever requests square images. All JPEG images must use the RGB or RGBA color space (CMYK is not supported), and be of the requested size (width, height). See Applying user records for more information. Default: <code>https://www.gravatar.com/avatar/{{mail trim lower pex_md5}}?d=404</code>

Option	Description
Allow the user's other personal details to be manually overridden	Allows the auto-generated user personal information (title, department and avatar URL) to be manually overridden for each user.
User advanced options (shown when Sync users is selected)	
UUID pattern	<p>The pattern to use to generate the UUID of the user.</p> <p>This field is required and must be unique for each user and must be in a UUID format. Therefore it is strongly recommended to use <code>{{objectGUID pex_to_uuid}}</code> as the value of this pattern.</p> <p>Default: <code>{{objectGUID pex_to_uuid}}</code></p>
Exchange mailbox UUID pattern	<p>The pattern to use to generate the Exchange Mailbox UUID of the user. This field is not required but if included it must be in a UUID format and be unique for each user.</p> <p>Default: <code>{% if msExchMailboxGuid %}{{msExchMailboxGuid pex_to_uuid}}{% endif %}</code></p>
Allow user advanced options to be manually overridden	Allows the auto-generated user advanced options to be manually overridden for each user.
Email options	
Send emails	<p>These email options allow you to send an email to the VMR owner whenever a synchronization creates a new VMR or modifies an existing VMR, or when a synchronization creates a new device alias or modifies an existing device alias.</p> <p>The VMR/Device owner's email address:</p> <ul style="list-style-type: none"> Defaults to <code>{{mail}}</code> Is used as the email address of the user records that are created when syncing users. When using the VMR self-service portal, it determines which VMRs can be viewed and edited (the LDAP mail attribute of the user logged into the VMR portal must match the VMR owner's email address). <p>See Sending provisioning emails to VMR and device owners for more information about these options.</p>
VMR / device owner's email address	
Address override	
SMTP server	

3. Select Save.

You must save the template before you can use it to generate VMRs.

Allowing generated fields to be manually overridden

Many of the settings can be tagged as "overridable" which means if that setting is manually overridden after the VMR, device or user has been created, a subsequent resynchronization of that template will not reset it back to another value.

For example, you could configure the template with **Allow PIN settings to be manually overridden** selected. Then, after generating a VMR with a random Host PIN, the administrator could manually change the PIN to another specific value. This "overridable" setting ensures that the new value is protected and is not reset to another random value when that template is next used to synchronize the VMRs.

If you make a field overridable you do not have to wait for another sync before making your override changes to the relevant field. To make a field modifiable via the template again, you must clear the relevant **Allow <field> to be manually overridden** template setting and then re-sync with that template. Note that a setting tagged as overridable via one template could still be modified by a different template.

All of the override options are disabled by default.

LDAP search and filter examples

You can configure multiple synchronization templates that all use the same LDAP data source (or different LDAP sources, if required), but that apply different filters to that LDAP source. This means that you could apply different VMR configuration patterns based on different LDAP organizational groups. For example, you could configure the VMRs for all members of the European sales team to have a Guest PIN and a different theme to those VMRs generated for the American accounting department.

If you want to apply different VMR patterns to different types of LDAP directory users, you can use the **LDAP user search DN** and **LDAP user filter** template fields to filter the records from the LDAP sync source that are used to generate VMRs via that template.

This section contains some examples that you could use in a Windows Active Directory environment. Note that all LDAP user search and user filter contents are not case sensitive.

Filtering based on group membership

To import users that belong to a specific group, you can filter on the **memberOf** AD attribute. For example, to only import users who are members of the `cn=sales,ou=groups,dc=example,dc=com` group, you would set the **LDAP user filter** on your template to:

```
(&(objectCategory=person)(objectClass=user) (memberOf=cn=sales,ou=groups,dc=example,dc=com))
```

You could then configure a second template for the accounting department that uses the same **LDAP sync source** but with the **LDAP user filter** set to:

```
(&(objectCategory=person)(objectClass=user) (memberOf=cn=accounting,ou=groups,dc=example,dc=com))
```

To only import users who are **not** a member of the sales group, you would set the **LDAP user filter** to:

```
(&(objectCategory=person)(objectClass=user) (!(memberOf=cn=sales,ou=groups,dc=example,dc=com)))
```

To import users that belong to a specific group and any other group **nested** within that group, you can use the **memberOf:1.2.840.113556.1.4.1941** filter for users. For example, to import users who are members of the `cn=sales,ou=groups,dc=example,dc=com` group or are members of a group nested within the sales group, you would set the **LDAP user filter** on your template to:

```
(&(ObjectCategory=user)(memberOf:1.2.840.113556.1.4.1941:=cn=sales,ou=groups,dc=example,dc=com))
```

Filtering by LDAP field names

To import only those users who have an email address, you can set the **LDAP user filter** to:

```
mail=*
```

To import all users **except** for some named individuals, you could use the following **LDAP user filter** model:

```
(&(objectCategory=person)(objectClass=user) (!(cn=Alice Parkes))(!(cn=Bob Lee))(!(cn=Carol Jones)))
```

Filtering by organizational unit (e.g. region/country)

To import users that are located in a specific organizational unit such as a region or country, you can restrict the user search to the appropriate 'ou' objects. For example, to only import users who are based in 'Europe', you would set the **LDAP user search DN** to:

```
ou=europe,ou=people
```

(this is the DN relative to the base DN defined in the LDAP sync source)

Escaping special characters in search/filter strings

If your search/filter strings include the following special characters: `()*[]\` you must use a single `\` to escape those characters.

For example if you have a group `cn=group-with-parenthese(s)-in-name,ou=groups,dc=example,dc=com` then you would set the **memberOf** filter to:

```
(memberOf=CN=group-with-parenthese\(s\) -in-name,ou=groups,dc=example,dc=com))
```

More information on Active Directory LDAP filtering can be found at

<https://social.technet.microsoft.com/wiki/contents/articles/5392.active-directory-ldap-syntax-filters.aspx>.

Generating / syncing VMRs, devices and users with LDAP

After you have configured a sync source and a template you can generate your VMRs, devices and users for the first time, or you can synchronize them to any changes that have occurred in the LDAP database since the last time the VMRs, devices and users were updated.

When synchronizing a template:

- Pexip Infinity checks all of the VMRs, devices and user records previously created by that template and it updates, deletes or creates new records as appropriate based on the current data in the sync source. Therefore, for example, if a user is removed from Active Directory or their account is disabled, then their corresponding VMR, device alias and user record will be deleted.
- If a generated VMR name, device alias or user email address already exists (either created manually or via a different template), or if a generated alias for a VMR is already assigned to another VMR, those existing VMR properties, VMR-to-alias assignments, device aliases or user records are not overwritten.

Any configuration changes made to Virtual Meeting Rooms are replicated to all Conferencing Nodes (typically after approximately one minute) and applied to any subsequent conferences in that Virtual Meeting Room. If there are any conferences already running that use the Virtual Meeting Room, any attempts to join it after the configuration has been replicated may be affected by the new configuration settings.

Changes to device aliases are also replicated to all Conferencing Nodes (again, typically after approximately one minute) and are applied to any subsequent registration requests.

The template synchronization process itself can also take several minutes if you have a large number of VMRs, devices or users. Therefore you must wait for at least one minute after the entire synchronization process has completed to ensure that all synced data has been replicated to all Conferencing Nodes.

For these reasons, we do not recommend changing Virtual Meeting Room configuration while a conference is in progress as this may lead to an inconsistent user experience.

Automatic synchronization

To automatically synchronize a template on a daily basis:

1. Go to **Utilities > LDAP Sync Templates**.
2. Select the template you want to use (to open the **Change LDAP Sync Template** page).
3. Select **Enable automatic daily sync**.
4. Select **Save**.

As template synchronization can result in the automatic creation, modification or deletion of large numbers of VMRs, devices and users, we recommend that you only enable automatic syncing after you have manually synced at least once and have verified that you are satisfied with your sync template configuration.

All automatic synchronizations are initiated at 01:00 UTC (this start time cannot be configured). After an initial sync, which can take several minutes in a large organization, an ongoing daily sync is typically much faster as it only processes changes since the previous sync.

Manual synchronization

To manually generate (for the first time) or synchronize the VMRs, devices and aliases in Pexip Infinity with the LDAP data source:

1. Go to **Utilities > LDAP Sync Templates**.
2. Select the template you want to use (to open the **Change LDAP Sync Template** page).
3. Scroll down to the bottom of the page and select **Sync now**.
4. A status page is displayed which shows the progress of the synchronization.
You can safely navigate away from this page without affecting the synchronization.
5. For large imports that take a long time to complete, you can monitor the ongoing [status](#) of the import at any time via **Status > LDAP Sync**.

Alternatively, you can sync one or more templates by going to **Utilities > LDAP Sync Templates**, selecting the check box next to the templates that you want to use, and then from the **Action** drop-down menu, select **Sync selected LDAP sync templates** and then select **Go**.

Deleting all VMRs, devices and users associated with a template

You can easily delete all of the VMRs, devices and users that have been created from a specific template, for example if you have incorrectly specified the template's LDAP user filter.

To delete all VMRs, devices and user records associated with a synchronization template:

1. Go to **Utilities > LDAP Sync Templates**.
2. Select the template whose associated VMRs, devices and users you want to delete (to open the **Change LDAP Sync Template** page).
3. Scroll down to the bottom of the page and select **Delete all VMRs, devices and users created from this template**.
4. Confirm that you want to proceed.
5. The VMRs, devices and users will be deleted and a status message is displayed confirming the number of records that were deleted.

Deleting a template

If you delete a synchronization template, all of the VMRs, devices and users associated with that template will also be deleted.

Applying user records

User records can be used to support the use of two features within Pexip Infinity: [participant avatars](#) and [scheduling meetings in personal VMRs](#).

User records can be [created manually](#) (Users & Devices > Users), or they can be [generated from directory information](#) contained in an AD/LDAP server via Pexip's provisioning mechanism (Utilities > LDAP Sync Templates).

User records and participant avatars

Conference participants and directory contacts within Pexip Infinity can be represented by an avatar or image.

You can configure user records to represent those participants/contacts and associate each user with an avatar URL that points to an external service (such as Gravatar) which can be used to retrieve that user's avatar/image. Then, when that user is participating in a conference, their avatar can be shown to any of the other conference participants who are using an Infinity Connect client.

When using avatars, in addition to creating the user record, you must also set up and enable an **Avatar policy** (Call Control > Policy Profiles).

When viewing participants' avatars within a conference:

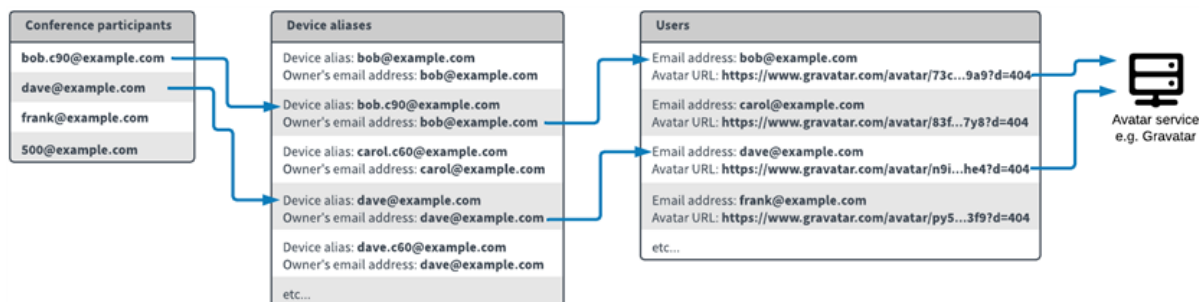
- For a participant's avatar to be requested and available for display, that participant's device must be registered to Pexip Infinity (any device type or protocol).
- If the participant joins as audio-only, their avatar is displayed to all of the video participants in the conference.
- Infinity Connect client users can see a participant's avatar via the participant list tab in the side panel and using the **Show Info** option.

Setting up user records and avatar URLs

When configuring end users and their associated avatars, there are two main attributes of each user record to consider:

- **Email address:** the user's **Email address** is used as the primary identifier of each user record. When attempting to retrieve a user's avatar, the system locates the relevant user record (and thus the user's avatar URL) by finding the user **Email address** that matches the **Owner's email address** associated with the [device alias](#) of the conference participant. We recommend using LDAP sync templates to provision the device and user records as this will ensure that a matching email address is used.
- **Avatar URL:** this is the link to where the avatar can be requested:
 - The avatar URL must be an unprotected resource (username and password credentials cannot be supplied with the request), and it must be reachable from Conferencing Nodes.
 - The image retrieved from the avatar URL must be a JPEG.
 - When a Conferencing Node sends an image request to the avatar URL, Pexip Infinity adds on extra URL parameters that specify the required dimensions, for example `?width=100&height=100&s=100` (the `s` is a size parameter used by Gravatar). Pexip Infinity only ever requests square images.
 - All JPEG images must use the RGB or RGBA color space (CMYK is not supported), and be of the requested size (width, height).

The following example diagram shows the relationship between participant aliases, device aliases and user records when obtaining an avatar URL:



To configure user records and avatar URLs:

1. Set up a [policy profile](#) with local avatar configuration enabled:
 - a. Go to **Call Control > Policy Profiles** and create a profile (or modify an existing profile).
 - b. In the **Avatar policy** section, enable **Use local avatar configuration**.
 - c. Save the policy profile.
 - d. Assign the policy profile to your locations (**Platform > Locations**).
2. Configure user records and their associated avatar URLs:
 - You can [bulk provision](#) the user records from directory information contained in a Windows Active Directory LDAP server, or any other LDAP-accessible database, or
 - You can manually configure user records (see [below](#)).

Note that other user attributes can also be configured (such as names and contact numbers) but these are not currently used within Pexip Infinity.

User records and personal VMR scheduling

Pexip Infinity's VMR Scheduling for Exchange feature can offer users the ability to [scheduling meetings in their own personal VMRs](#). This feature requires users to sign in to an Outlook add-in using their SSO email address, and requires Pexip Infinity to be configured with both:

- a user record with an **Email address** that matches the user's SSO email address (usually the user record will already exist, but you can configure this feature to create the required user record if required, in which case the [User origin](#) will display the name of the associated VMR scheduling for Exchange integration), and
- at least one VMR with an **Owner's email address** that matches the user's SSO email address.

Manually configuring user records

To manually configure the user records, go to **Users & Devices > Users**. The full set of options are:

Option	Description
Email address	The email address of the user. If you wish to use Personal VMRs, this must be the user's SSO email address.
Description	The description of the user.
First name	The first name of the user.
Last name	The last name of the user.
Display name	The display name of the user. Note that the display name is not currently used in Pexip Infinity (for example, it does not affect participant name overlays and cannot be used when provisioning Infinity Connect clients).
Telephone number	The telephone number of the user.
Mobile number	The mobile number of the user.
Title	The job title of the user.
Department	The department of the user.
Avatar URL	The avatar URL of the user.
Advanced options	

Option	Description
UUID	<p>The unique identifier (UUID) of the user.</p> <p>This field is required and must be unique for each user and must be in a UUID format. Therefore it is strongly recommended to use the generated default value.</p>
Exchange mailbox UUID	<p>The unique identifier of the user's Exchange Mailbox. This field is not required but if included it must be in a UUID format and be unique for each user.</p>
User origin	<ul style="list-style-type: none"> If the user was provisioned from Active Directory, this is the name of the LDAP sync template used to create this user. If the user was created as part of scheduling meetings in personal VMRs, this is the name of the VMR scheduling for Exchange integration that created the user. If the user was created by manual input or via the API, this will be blank. <p>This field is read-only.</p>

Using templates, variables and filters when provisioning VMRs, devices and users

When you configure an LDAP synchronization template to [provision VMRs, devices and users](#), you define patterns for how some of those properties — such as the VMR name or a device alias — are generated from the data in the LDAP sync source.

Pexip Infinity's LDAP sync templates use a subset of the jinja2 templating language (<https://jinja.palletsprojects.com/en/2.10.x/templates/>).

These templates consist of the following elements:

- **literal text** that you want to add to the output or result, such as prefixing every generated VMR name with `meet`.
- **variables** that are substituted with values from the LDAP sync source, such as `givenName`
- **filters** that can manipulate text or modify the content of variables or text strings, such as `join` or `lower`
- **delimiters** such as `{{...}}` and pipes `|` which are used to enclose variables and define filter expressions
- **jinja statements and control structures** (see <https://jinja.palletsprojects.com/en/2.10.x/templates/#list-of-control-structures>)

An example pattern would be `meet.{{givenName|lower}}.{{sn|lower}}`. This concatenates the literal text `meet`. with the content of the `givenName` variable that has been converted to lower case by the `lower` filter. This is then concatenated with a period `.` and then the `sn` (surname) variable, also piped through the `lower` filter.

Therefore if Alice Parkes had an entry in the LDAP directory with `givenName=Alice` and `sn=Parkes`, the example pattern above would produce `meet.alice.parkes`.

The following sections provide more information about the supported [variables](#), [jinja2 filters](#) and [custom Pexip filters](#), with each section including example expressions that demonstrate how to format your patterns.

Supported variables

When syncing each VMR, device or user record, Pexip Infinity extracts LDAP fields from the directory records in the LDAP data source and makes them available as variables (with the same name) for use in the sync template. Pexip Infinity offers several [standard](#) fields, and administrators can also add their own [custom](#) set of fields. Fields with [multiple values](#) are also supported.

When using variables:

- You must enclose the variable name within `{{...}}` delimiters, for example `{{givenName}}`.
- Any combination of the variables can be used in any synchronization template pattern field.
- Fields that are not present in the LDAP source will appear as "None".
- Capitalization of the variable name is important. The variable **must** be spelled exactly as shown below.

Standard LDAP fields

Pexip Infinity automatically extracts and makes available the following standard LDAP fields:

Variable name	Description
company	Company/organization name
department	Department name
displayName	User's preferred display name
employeeID *	Employee reference number
givenName	First name
mail	Email address
mailNickname *	Typically used to store an alternative email address (e.g. based on a maiden name); see Change of name below
mobile	Mobile phone number
objectGUID	Globally unique identifier for the directory object (user entry)
sAMAccountName	Logon name (domainname\username format)
sn	Last name or surname
telephoneNumber	Telephone number
title	Job title
userPrincipalName	Logon name (username@domainname format)
* Not present in all AD schemas.	

Custom LDAP fields

In addition to the standard set of fields made available automatically by Pexip Infinity, administrators can configure their own custom set of fields to support additional attributes in their LDAP/AD schemas.

To add a custom LDAP field:

1. Go to **Utilities > LDAP Sync Fields**.
2. Select **Add LDAP sync field**, and then complete the following fields:

LDAP field name	The name of LDAP field to be read from the LDAP server. The name is case sensitive. If the name does not match exactly against a field in the LDAP data source its value will appear (when used in a template) as "None".
Template variable name	The name of the variable to use in sync templates that will contain the value of the LDAP field. We recommend that you set this name to something similar to the LDAP field name, but note that this name cannot contain hyphens.
Description	An optional description of the LDAP attribute.
Is binary (advanced options)	In advanced scenarios, some binary LDAP fields such as GUIDs require special encoding. In such cases, expand the advanced options and select Is binary . Do not select this option for ordinary textual or numeric LDAP fields.

3. Select **Save**.

The **Template variable name** that represents the custom LDAP field can now be used in a template along with the standard LDAP fields.

LDAP fields with multiple values

In some scenarios, an LDAP field could contain multiple values for a given user. For example the LDAP `proxyAddresses` field (which you would have to configure as a custom LDAP field as described above) could contain many addresses.

In these cases, a special `multi_valued_attrs` variable is automatically created. This variable will contain **all** of the LDAP fields that are found to have multiple values for a given user. This means if two fields have multiple values, the `multi_valued_attrs` variable will contain all of the values for those two fields. It will not contain any of the fields that have just a single value.

You can then use the `pex_find_first_match` filter to extract values from the variable. This filter can be used to extract from a string list the first value that matches a specified regex.

Example usage: `{{pex_find_first_match(multi_valued_attrs.proxyAddresses, "^sip:.*)}}`

This would extract the first `proxyAddresses` value that starts with "sip:" i.e. the first proxy that has a SIP-style address.

However, as the `multi_valued_attrs` variable is only populated with LDAP fields that contain multiple values, a better expression in this case would be: `{{pex_find_first_match(multi_valued_attrs.proxyAddresses, "^sip:.*) or proxyAddresses}}` which will return the first/only value in the standard `proxyAddresses` variable if only one value is available or the regex does not find a match.

More complex matching is possible by iterating over the `multi_valued_attrs` collection in other ways.

Change of name

A typical scenario encountered by IT administrators is when someone changes their name (e.g. after getting married) and wants to preserve their original contact address alongside their new contact details.

One way in which you could support this in Pexip Infinity is by defining multiple VMR alias patterns in your template. This example approach assumes that the LDAP `mail` field holds the user's contact address, and a `mailNickname` field is used to hold an alternative address. Thus, you could define the following patterns:

VMR alias 1 pattern: `meet.{{mail}}`

VMR alias 2 pattern: `{%if mailNickname %}{#Add alias based on user's maiden name mailNickname#}meet.{{mailNickname}}{% else %}{#intentionally leave the alias blank#}{% endif %}`

For most users, this will generate a single VMR alias in the format `meet.<email address>`, and the VMR's second alias will remain blank as the LDAP `mailNickname` field for those users will be empty.

For example, user Ann Jones has her LDAP `mail` field set to `ann.jones@example.com` and her LDAP `mailNickname` field is blank. Her VMR will have a single alias of `meet.ann.jones@example.com`.

When a user changes their name, the administrator would update the user's LDAP `mail` field to match their new name and populate the `mailNickname` field with the previous value of the `mail` field.

Now, the next time the administrator performs a template synchronization, the user's VMR aliases will be updated. The first alias will be changed to a new string based on the user's new email address, and a second alias will also be generated based upon the `mailNickname` field. This means that the user can be contacted via either their previous VMR alias or the new alias.

In our example, let's assume that Ann Jones changes her name to Ann Smith. The administrator changes her LDAP `mail` field to `ann.smith@example.com` and sets her LDAP `mailNickname` field to `ann.jones@example.com`. Her VMR will now have two aliases: `meet.ann.smith@example.com` and `meet.ann.jones@example.com`.

Supported jinja2 filters

Pexip Infinity supports a subset of filters from the jinja2 templating language. Any jinja filters that are not listed below have been disabled in Pexip Infinity. See <https://jinja.palletsprojects.com/en/2.10.x/templates/#list-of-builtin-filters> for more information about these filters.

abs	float	last	replace	truncate
capitalize	format	length	round	upper
default	int	lower	striptags	
first	join	range	trim	

To use a filter you would typically follow the syntax `{{<source_value>|<filter_name>}}`.

In most cases the `<source_value>` is likely to be a variable, for example `{{givenName|upper}}`, although it could be one or more literal values, for example `{{ [1, 2, 3, 4]|join }}`.

Some filters take parameters, for example `{{sn|truncate(5)}}`. You can also use multiple filters in the same expression, for example `{{sn|truncate(5)|upper}}`.

The `trim` filter is often used. This trims leading and trailing whitespace from the string held in the `<source_value>`.

Example usage: `{{title|trim}}`

If the `title` field contained " Project Manager ", this would be converted to "Project Manager".

The `replace` filter is also often used. This replaces one string with another string. The first argument of the filter is the substring that should be replaced, the second is the replacement string. If the optional third argument `count` is given, only the first count occurrences are replaced.

Example usage: `{{ department|replace("Personnel", "HR") }}`

If the `department` field contained "Personnel Department Personnel", this would be converted to "HR Department HR".


Example usage: `{{ department|replace("Personnel", "HR", 1) }}`


In this case a count of 1 is specified, thus if the `department` field contained "Personnel Department Personnel", it would be converted to "HR Department Personnel".

For more complicated search and replace patterns, use the custom Pexip `pex_regex_replace` filter described below.

Custom Pexip filters

In addition to the Jinja filters, Pexip also provides the following custom filters, which are typically used to manipulate data:

Filter	Description and example usage
<code>pex_base64</code>	<p>Performs Base64 encoding on the input field.</p> <p>Example usage: <code>{{provisiondata pex_base64}}</code></p> <p>In this example, a local variable named <code>provisiondata</code> is encoded as Base64.</p>
<code>pex_clean_phone_number</code>	<p>This extracts only +0123456789 characters (and removes ()&%#@ '";:, A-Z,a-z etc).</p> <p>Example usage: <code>{{ telephoneNumber pex_clean_phone_number }}</code></p> <p>In this example, if <code>telephoneNumber</code> is '+44 (20) 12345678', this expression would return '+442012345678'.</p>
<code>pex_debug_log(message)</code>	<p>The <code>pex_debug_log</code> filter can be used to help debug your script. It writes debug messages to the Pexip Infinity support log. You can include literal text and variables.</p> <p> To avoid filling the support log and causing it to rotate, remove all <code>pex_debug_log</code> filters from your scripts as soon as they are working correctly.</p>
<code>pex_find_first_match(string_list, 'find_regex')</code>	<p>This extracts from the list the first value that matches the specified regex.</p> <p>It is typically used to extract a value from the <code>multi_valued_attrs</code>variable where an LDAP field, such as <code>proxyAddresses</code>, contains multiple values.</p> <p>Example usage: <code>{{pex_find_first_match(multi_valued_attrs.proxyAddresses, "^sip:.*)}}</code></p> <p>This would extract the first <code>proxyAddresses</code> value that starts with "sip:" i.e. the first proxy that has a SIP-style address.</p>
<code>pex_hash</code>	<p>Performs a hash of a field.</p> <p>You could use this, for example, to generate random-looking PINs based on a hash of one or more fields from the directory (you would also need to apply another filter such as <code>pex_tail</code> to set the PIN to an appropriate valid length).</p> <p>Example usage: <code>{{ sAMAccountName pex_hash pex_tail(6) }}</code></p>

Filter	Description and example usage
pex_head (maxlength)	<p>Returns, at most, the first maxlength characters from the input field.</p> <p>Example usage: <code>{{ givenName pex_head(4) }}</code></p> <p>In this example, for a givenName of 'Alice' this expression would return 'Alic', and a givenName of 'Bob' would return 'Bob'.</p>
pex_in_subnet	<p>Tests whether a given address is within one or more subnets. It takes as input the address you want to test, and one or more subnet ranges, and returns either True or False.</p>
pex_md5	<p>Applies an MD5 hash to the input field.</p> <p>Example usage: <code>https://www.gravatar.com/avatar/{{mail trim lower pex_md5}}?d=404</code></p> <p>This example generates a URL that conforms with how the Gravatar service constructs its URLs based on users' email addresses.</p>
pex_now (timezone)	<p>The <code>pex_now</code> filter takes an optional parameter of a timezone description e.g. 'UTC', 'Asia/Tokyo', or 'US/Eastern' and returns the current date and time for that timezone. UTC is assumed if a timezone is not provided.</p> <p>The resulting available attributes are <code>year</code>, <code>month</code>, <code>day</code>, <code>hour</code>, <code>minute</code>, <code>second</code> and <code>microsecond</code>.</p> <p>Example usage:</p> <pre>{% set now = pex_now("Europe/London") %} {% if now.month == 2 and now.day == 29 %}</pre>
pex_random_pin(length)	<p>Generates a random PIN of the given length. Note that this filter does not take any input.</p> <p>Example usage: <code>{{ [9,pex_random_pin(5)] join }}</code> for the Host PIN, and <code>{{ [2,pex_random_pin(5)] join }}</code> for the Guest PIN.</p> <p>A usage pattern such as this (where the Host PIN starts with a 9, and the Guest PIN starts with a 2) ensures that a VMR can never be configured with a Host PIN that is the same as the Guest PIN.</p> <p>You would typically use this filter in conjunction with the Allow PIN settings to be manually overridden option to ensure that the PIN is not reset every time a template resync is performed.</p> <p> Do not use <code>pex_random_pin()</code> to generate aliases. This filter generates a truly random number, with each number generated independently of any previous output. Therefore, with many thousands of users, a 5 digit numeric alias (e.g. <code>pex_random_pin(5)</code>) is statistically quite likely to clash with a random alias generated using <code>pex_random_pin(5)</code> for another user. With 10,001 or more users and a 4 digit random alias, a clash is guaranteed (with multiple clashes being likely).</p>
pex_regex_replace('find_regex', 'replace_string')	<p>This performs a regex find and replace.</p> <p>Example usage: <code>{{mail pex_regex_replace('@.+','@otherdomain.com')}}}</code></p> <p>This example takes as input an email address contained in the <code>mail</code> variable and changes the domain portion of the address to <code>@otherdomain.com</code>. For example, it will transform <code>user1@domainA.com</code> to <code>user1@otherdomain.com</code>, and <code>user2@domainB.com</code> to <code>user2@otherdomain.com</code> etc.</p>
pex_regex_search('regex_pattern', 'string_to_search')	<p>This performs a regex search for the first location that matches the pattern and returns the regex groups.</p> <p>Example usage:</p> <pre>{% set groups = pex_regex_search("[a-z0-9-]+@[a-z0-9-]+.com)", "example string with someone@example.com") %} {% if groups %} {{ groups[0] }}@{{ groups[1] }} % endif %</pre> <p>This example takes as input a string containing an email address and extracts the email using two regex groups.</p>

Filter	Description and example usage
pex_require_min_length (length)	<p>This validates that the input string field has the specified minimum length.</p> <p>Syntax: <code>{{ some_string pex_require_min_length(2) }}</code></p> <p>You can use this filter to control if VMRs are created or not, based on the length of a string or variable.</p> <p>For example, you could set the VMR description pattern as <code>{{ mail pex_require_min_length(1) }}</code>. This means that the VMR will not be created if the mail variable is empty. (When subsequently syncing, the VMR would be deleted if the condition is not met.)</p> <p>Example usage: <code>{{ telephoneNumber pex_require_min_length(4) }}</code></p> <p>This example ensures that a VMR is created only if <code>telephoneNumber</code> is at least 4 characters long.</p>
pex_reverse	<p>This reverses the characters in the input field.</p> <p>Example usage: <code>{{ givenName pex_reverse lower }}</code></p> <p>In this example, if <code>givenName</code> is 'Alice', this expression would return 'ecila'.</p>
pex_strlen	<p>Returns the length of string. The basic usage syntax is:</p> <pre>{% set some_length = "Example" pex_strlen %} {# sets a variable named some_length to the value 7 #}</pre> <p>Example usage: <code>{%set a = telephoneNumber pex_clean_phone_number pex_head(8)%}%set b = mobile pex_clean_phone_number pex_head(8)%}%if a pex_strlen== 8%}{a}%else%}{b}%endif%</code></p> <p>In this example, the expression returns the value of the <code>telephoneNumber</code> field if it is present and at least 8 characters long after cleaning and truncation, otherwise it returns the value of the <code>mobile</code> field that has been cleaned and truncated to at most 8 characters.</p> <p>Example usage: <code>{%if telephoneNumber pex_strlen == 5 %}{#return user's phone number#}{{telephoneNumber pex_clean_phone_number}}{% else %}{#intentionally leave blank#}{% endif %}</code></p> <p>In this example, the expression returns a cleaned value of the <code>telephoneNumber</code> field if it is 5 characters long, otherwise it returns an empty string.</p>
pex_tail (maxlength)	<p>Returns, at most, the last <code>maxlength</code> characters from the input field.</p> <p>Example usage: <code>{{ telephoneNumber pex_tail(4) }}</code></p> <p>In this example, if <code>telephoneNumber</code> is '+44 (20) 12345678', this expression would return '5678'.</p>
pex_to_json	<p>Converts a Python dictionary variable into JSON format.</p>
pex_to_uuid	<p>Converts a base64 string to a UUID.</p> <p>GUIDs that are retrieved from LDAP are encoded as base64 and typically need converting to a human readable UUID such as a59da36d-9b16-430a-80f7-cb1b01d4bd45 for subsequent use within Pexip Infinity.</p> <p>Example usage: <code>{{objectGUID pex_to_uuid}}</code></p>
pex_update	<p>Updates Python dictionary variables.</p>
pex_url_encode	<p>This filter creates URL parameters that are safely URL-encoded.</p> <p>It takes any number of two-element tuples and converts them into a percent-encoded string of key=value pairs. It does not take any input; the syntax is:</p> <pre>{{ pex_url_encode(('key_1', 'data_1'), ('key_2', 'data_2'), ..., ('key_n', 'value_n')) }}</pre> <p>Example usage: <code>{{ pex_url_encode(('data', '8J+UpckvXF8o44OEKV8vwq/wn5SI'), ('message', 'A test message')) }}</code> which would produce: <code>data=8J%2BUpcKvXF8o44OEKV8vwq%2Fwn5SI&message=A+test+message</code></p>
pex_uuid4()	<p>This generates a uuid (universally unique identifier). Note that this filter does not take any input.</p> <p>Example usage: <code>{{ pex_uuid4() }}</code></p> <p>In a similar manner to the <code>pex_random_pin</code> filter, you would typically use this in conjunction with the appropriate <code>Allow <field> to be manually overridden</code> template setting, to ensure that after a uuid has been assigned to a field, another different uuid is not assigned every time a template resync is performed.</p>

Sending provisioning emails to VMR and device owners

When [bulk-provisioning VMRs and device aliases](#) from Active Directory via LDAP, an email can be sent out to the VMR owner or device owner telling them the:

- VMR properties, such as its aliases and security PINs
- device alias properties, such as its associated username and password for registration purposes.

Email generation is specified on a per sync template basis. The content of the email is free-form and can be customized for each generated VMR or device alias by using variables — such as {{pin}} to include the Host PIN in VMR-related emails — that are substituted with the relevant value when each individual email is generated.

This topic covers:

- [Guidelines and limitations](#)
- [Configuring an LDAP sync template to generate emails](#)
- [Constructing the VMR provisioning email](#)
- [Example VMR email templates](#)
- [Constructing the device alias provisioning email](#)
- [Example device email templates](#)
- [Sending reminder emails](#)

Guidelines and limitations

When constructing your email template:

- Ensure that the subject line of the generated email is a single line.
- You can use HTML markup (UTF-8 characters only).

If Pexip Infinity experiences connectivity issues with the SMTP server, it will retry sending each individual email up to 3 times before moving onto the next email, and it will attempt to re-establish dropped SMTP server connections up to 100 times per bulk-send operation.

Note that an email is triggered when a synchronization process results in changes to the VMR's or device alias's properties, regardless of whether that property is available as an email variable, and whether that variable is actually used in an email template. Changes to the email template do not trigger the sending of an email.

If you want to update all VMRs, for example with a new layout, but do not want to trigger emails, then disable the **Send emails** option for the sync template, make the change to the template, synchronize, and when completed, turn **Send emails** back on again.

Configuring an LDAP sync template to generate emails

When configuring a sync template (**Utilities > LDAP Sync Templates**) you can specify the following email-related options:

Email options	
Send emails	<p>When selected, the system generates and sends an email to the:</p> <ul style="list-style-type: none">• VMR owner when a synchronization creates a new VMR or modifies an existing VMR (when Sync VMRs is enabled for the template)• device owner when a synchronization creates a new device alias or modifies an existing device alias (when Sync devices is enabled for the template). <p>Separate emails are sent for VMR provisioning and for device provisioning when both Sync VMRs and Sync devices are enabled. Note that you can also manually send reminder emails.</p>

Email options	
VMR / device owner's email address	<p>The email address of the owner of this VMR or device alias. The generated email(s) will be sent to this address.</p> <p>This field is also used as the email address of the user records that are created when Sync users is enabled.</p> <p>When using the VMR self-service portal, it determines which VMRs can be viewed and edited (the LDAP mail attribute of the user logged into the VMR portal must match the VMR owner's email address).</p> <p>Example: {{mail}}</p>
Allow email address to be manually overridden	<p>Allows the auto-generated email address to be manually overridden for each VMR or device alias.</p>
SMTP server	<p>Select the SMTP server to use for sending provisioning emails (see Configuring SMTP servers).</p>
VMR email subject template	<p>A template for the subject line of the email to be sent when a VMR is created or updated.</p>
VMR email body template	<p>A template for the body of the email to be sent when a VMR is created or updated.</p> <p>See Constructing the VMR provisioning email below for more information on how complete this field.</p>
Device email subject template	<p>A template for the subject line of the email to be sent when a device alias is created or updated.</p>
Device email body template	<p>A template for the body of the email to be sent when a device alias is created or updated.</p> <p>See Constructing the device alias provisioning email below for more information on how complete this field.</p>

Constructing the VMR provisioning email

The **VMR email subject template** and **VMR email body template** fields contain the pattern for the email to be sent to the VMR owner when that VMR is first created (provisioned) or is updated.

The templates will typically contain a mixture of literal text and variables, and it can also contain control structures that allow you to vary the content of the email based upon certain conditions.

Supported variables

The **VMR email subject template** and **VMR email body template** fields support a different set of variables from the other synchronization template pattern fields (because it is based on the generated VMR properties, rather than the source data used to build those properties).

The following variables are available:

Variable name	Description
primary_owner_email_address	The email address of the owner of the VMR.
description	The description of the VMR.

Variable name	Description
aliases	<p>A list of tuples containing the VMR alias and its corresponding description e.g. <code>[("alice", "The short alias"), ("meet.alice@example.com", "The alphanumeric URI alias"), ("123456", "The numeric alias")]</code></p> <p>Thus, in your email template, <code>aliases[0][0]</code> will extract the first alias in the aliases list, <code>aliases[0][1]</code> extracts the description of the first alias in the list, and <code>aliases[1][0]</code> extracts the next alias in the list, and so on.</p> <p>Note that:</p> <ul style="list-style-type: none"> The order of the aliases contained in the <code>aliases</code> variable may not reflect the order of the alias templates in the sync template, or the order that the aliases appear when viewing the VMR via the Administrator interface or via the management API i.e. the alias generated by VMR alias 1 pattern may not appear before the alias generated by VMR alias 2 pattern and so on. As a VMR alias pattern template could generate a blank alias (which will be discarded when syncing) there is no guarantee that you will have 4 aliases configured even if a sync template has VMR alias 1, 2, 3 and 4 patterns configured. <p>Extracting a specific alias</p> <p>There are several ways in which you can extract a specific alias from the <code>aliases</code> variable. Let's assume that <code>aliases</code> contains the following (in any order):</p> <pre>[("alice", "The short alias"), ("meet.alice@example.com", "The alphanumeric URI alias"), ("123456", "The numeric alias")]</pre> <p>To extract an alias with a particular description, you can use this jinja code and text in your VMR email template:</p> <pre>{% macro _aliases_with_description(description) -%} {%for alias in aliases%}{%if alias[1]==description%}{alias[0]}{%endif%}{%endfor%} {%- endmacro %} The telephone number is {{_aliases_with_description("The numeric alias")}} The alphanumeric URI is {{_aliases_with_description("The alphanumeric URI alias")}}</pre> <p>and this will result in an email containing:</p> <pre>The telephone number is 123456 The alphanumeric URI is meet.alice@example.com</pre> <p>Alternatively, if you want to look at a start and end pattern of the alias itself to see if it is the alias you want to use, you can use this jinja code and text in your VMR email template:</p> <pre>{% macro _aliases_with_start_and_end(aliasstart, aliasend) -%} {%for alias in aliases%}{%if alias[0].startswith(aliasstart) and alias[0].endswith(aliasend)%}{alias[0]}{%endif%} {%endfor%} {%- endmacro %} The URI is {{_aliases_with_start_and_end("meet.", "@example.com")}}</pre> <p>which in this example only lists those aliases that start with <code>meet.</code> and end with <code>@example.com</code> i.e. it will result in an email containing:</p> <pre>The URI is meet.alice@example.com</pre> <p>Alternatively, you can use the system-generated <code>uri_alias</code> and <code>numeric_alias</code> variables as described below.</p>
allow_guests	<p>Whether to distinguish between Host and Guest participants:</p> <ul style="list-style-type: none"> <code>true</code> — the conference has two types of participants: Hosts and Guests. <code>false</code> — all participants have Host privileges
pin	The Host PIN.
guest_pin	The Guest PIN.
max_callrate_in	The maximum inbound call bandwidth (kbps).
max_callrate_out	The maximum outbound call bandwidth (kbps).

Variable name	Description
participant_limit	The maximum number of participants allowed to join this VMR.
tag	The VMR's service tag.
Additional generated variables	
action	Possible values are: <ul style="list-style-type: none"> "created" — when the VMR has just been created "updated" — when the existing VMR has been updated "reminder" — when sending reminder emails.
uri_alias	Set to any one of the configured URI aliases of a VMR. For example, <code>meet.alice@example.com</code> is a possible URI alias, as is <code>123456@example.com</code> . Thus if a VMR has both of those aliases then either might appear in the <code>uri_alias</code> variable. (Note that <code>123456</code> is not a URI alias.) If your VMRs have several URI-style aliases, you may want to programmatically select the alias that meets your required pattern from the <code>aliases</code> variable as described above . It is blank if there is no URI-style alias.
numeric_alias	Set to any one of the VMR's all-numeric aliases i.e. an alias consisting entirely of digits 0-9. It is blank if there is no numeric alias.

For more information about how participants join conferences, see [Creating preconfigured links to conferences via Infinity Connect](#).

Example VMR email templates

Here are some example email body templates that you can use as the basis for your own emails.

Basic VMR email template

This is a basic email template that can be used to show joining instructions for VMRs that do not have a PIN.

It makes use of the `{{numeric_alias}}` and `{{uri_alias}}` variables which, when the email is generated, will be substituted with the appropriate alias values for that specific VMR.

You can remove the instructions for any of the joining methods that are not appropriate for your deployment, such as the instructions for joining via the Infinity Connect client or via telephone. If you use this example, remember to replace `node.example.com` with the address of your Conferencing Node, and to use your own direct dial telephone number instead of `+1 555 0123`.

```

Here are the details of your Pexip Virtual Meeting Room (VMR). You can copy and paste these instructions when inviting people to join your meetings.

From a web browser, go to: https://node.example.com/webapp/#/?conference={{numeric_alias}} then enter your Name and select "OK"

From a video conferencing endpoint dial {{uri_alias}}

To join from an Infinity Connect client, click pexip://{{uri_alias}} (you can download it from https://www.pexip.com/apps or from the app store for your iOS/Android device)

To join from a telephone, dial +1 555 0123 then, when prompted, enter the conference number {{numeric_alias}} followed by "#"

Instructions for using your VMR can be found at https://docs.pexip.com/admin/connect_quick.htm

You can check your connection and the quality of your video and audio - from a web browser, go to:
https://node.example.com/webapp/conference/test_call

```

Constructing the device alias provisioning email

The **Device email subject template** and **Device email body template** fields contains the pattern for the email to be sent to the device owner when that device alias is first created (provisioned) or is updated.

The templates will typically contain a mixture of literal text and variables, and it can also contain control structures that allow you to vary the content of the email based upon certain conditions.

Supported variables

The **Device email subject template** and **Device email body template** fields support a different set of variables from the other synchronization template pattern fields (because it is based on the generated device properties, rather than the source data used to build those properties).

The following variable are available:

Variable name	Description
primary_owner_email_address	The email address of the owner of the device alias.
device_alias	The alias of the device that can be registered to Pexip Infinity.
device_description	The description of the device alias.
device_username	The username associated with the device alias. This should be used in association with the device_password when registering the device_alias to Pexip Infinity.
device_password	The password associated with the device alias.
device_tag	The device alias's service tag.
Additional generated variables	
action	Possible values are: <ul style="list-style-type: none">"created" — when the device alias has just been created"updated" — when the existing device alias has been updated"reminder" — when sending reminder emails.

Example device email templates

Here is an example email body template that you can use as the basis for your own emails. It is shown in plain text and in HTML format.

Basic device email template in plain text

This is a basic email template that can be used to provide the information required for registering a device.

It supplies the credentials (**device_username** and **device_password**) that must be used when registering the **device_alias** to Pexip Infinity.

If you use this example, remember to replace `confnode.example.com` with the address of your Conferencing Node.

```
Hi {{primary_owner_email_address}},

{%if action=="created" %}Your Infinity Connect client account has just been created.{% else %}Service update: your Infinity Connect client
account has been updated with new settings. {%endif%}

This email contains the settings to use when registering your Infinity Connect client (or your traditional hardware based video endpoint).
```

```
Alias: {{device_alias}}

Server address: confnode.example.com

User name: {{device_username}}

Password: {{device_password}}

When your device is registered, other users will be able to call you using your personal video alias: {{device_alias}}

Instructions for registering your device can be found at https://docs.pexip.com/clients/registering.htm
```

Basic device email template in HTML format

This is a repeat of the basic email template above, but illustrates how HTML formatting may be applied.

```
<html>
<p>Hi {{primary_owner_email_address}},</p>
<p>{%if action=="created" %}Your Infinity Connect client account has just been created.{% else %}Service update: your Infinity Connect client
account has been updated with new settings. {%endif%}</p>
<p>This email contains the settings to use for your Pexip Infinity Connect client (or your traditional hardware based video endpoint).</p>
<p>To configure your Pexip infinity client (or hardware device) you will need to enter the following settings:</p>
<p>Alias: <b>{{device_alias}}</b></p>
<p>Server address: <b>confnode.example.com</b></p>
<p>User name: <b>{{device_username}}</b></p>
<p>Password: <b>{{device_password}}</b></p>
<p>When your device is registered, other users will be able to call you using your personal video alias: {{device_alias}}</p>
<p>Instructions for registering your device can be found on the <a href="https://docs.pexip.com/clients/registering.htm" target="_blank">Pexip
documentation website</a></p>
</html>
```

If you are provisioning Infinity Connect users with their registration details, see [Registering and provisioning the Infinity Connect client](#) for some more examples of provisioning email template content.

Sending reminder emails

You can manually send reminder emails for a specific VMR or device alias, or for all VMRs and devices associated with a specific sync template. Note that:

- No status information is shown on the **LDAP Sync Status** page when sending email reminders (as the reminder process does not resync the content from the LDAP data source). However, all related activity is recorded in the administrator log as usual.
- You can only resend emails for VMRs or device aliases that have been created by an LDAP sync template.

Sending a reminder email to a VMR owner

To send a reminder email to the owner of a VMR:

1. Go to **Services > Virtual Meeting Rooms**.
2. Select the Virtual Meeting Room(s) for which you want to send reminders (select the checkbox next to the VMR name).
3. From the **Action** drop-down list, select **Send reminder emails to VMR owners**.
4. Select **Go**.

This will generate and send an email to the owner of each selected VMR.

To see or change the "owner" of a VMR, go into the VMR details page and show the **Advanced options**. The **Owner's email address** field is at the bottom of that section.

Sending a reminder email to a device owner

To send a reminder email to the owner of a device alias:

1. Go to **Users & Devices > Device Aliases**.
2. Select the device aliases for which you want to send reminders (select the checkbox next to the device alias name).

3. From the **Action** drop-down list, select *Send reminder emails to device owners*.
4. Select **Go**.

This will generate and send an email to the owner of each selected device.

To see or change the "owner" of a device, go into the device alias details page and review the **Owner's email address** field.

Sending reminder emails for VMRs and devices associated with a sync template

This option is useful if you have initially configured your system not to send emails — for example while you are ensuring that the VMRs and devices are being created as expected — and then want to send emails to all of those generated VMRs and devices associated with that template (as the synchronization process only automatically generates emails for an existing VMR or device if its configuration changes in some way).

To send reminder emails for all VMRs and devices associated with a specific sync template:

1. Go to **Utilities > LDAP Sync Templates**.
2. Select the sync template(s) for which you want to send reminders (select the checkbox next to the template name).
3. From the **Action** drop-down list, select *Send VMR and device reminder emails for selected LDAP sync templates*.
4. Select **Go**.

This will generate and send an email to the owner of each VMR and device that is associated with that LDAP sync template.

Troubleshooting LDAP server connections

Pexip Infinity can be configured to connect to a Windows Active Directory LDAP server, or any other LDAP-accessible database, in order to:

- [bulk-provision individual Virtual Meeting Rooms or devices](#) for every member of the directory
- [authenticate and authorize the login accounts](#) that are allowed to connect to the Pexip Infinity Administrator interface or the Pexip Infinity API.

This section explains how Pexip Infinity connects to the LDAP server, and provides guidance on how to troubleshoot connection problems.

Note that all LDAP distinguished names must be entered as per the LDAP standard ([RFC 4514](#)). LDAP configuration is case insensitive.

Connecting to the LDAP server

When resolving the **LDAP server address**, the system supports DNS SRV and DNS A/AAAA lookups. The system always tries in the first instance to set up a TLS connection with the LDAP server. If that fails it may fall back to a TCP connection if allowed.

To establish a TLS connection, the Pexip Infinity platform must trust the certificate presented by the LDAP server i.e. the LDAP server's certificate must be signed by an authority within the Pexip Infinity trusted CA certificates store. In addition, the resolved LDAP server address must match the CN (common name) contained within the certificate presented by the LDAP server.

The system will connect to the port returned by an SRV lookup, otherwise it will connect to 389 (TCP) or 636 (TLS). Requests to search the Active Directory Global Catalog use ports 3268 (TCP) and 3269 (TLS).

Connection process

If the **LDAP server address** is configured as an IP address, the system will connect directly to the given address, otherwise it treats it as a domain or FQDN and attempts to resolve the address via DNS lookups in the following sequence:

1. Perform a DNS SRV lookup against `_ldaps._tcp.<LDAP server address>`
(or `_ldaps._tcp.gc._msdcs.<LDAP server address>` if searching the AD global catalog).
2. Perform a DNS SRV lookup against `_ldap._tcp.<LDAP server address>`
(or `_ldap._tcp.gc._msdcs.<LDAP server address>` if searching the AD global catalog).
3. Perform a DNS A/AAAA lookup against `<LDAP server address>`.

When a DNS lookup is successful, the system will first attempt to establish a TLS connection with the server at the returned address. If the TLS connection attempt fails, the system will then attempt a TCP connection, but only if **Allow insecure transport** is enabled. Only TLS connections are attempted as a result of `_ldaps` lookups.

If multiple addresses are returned by SRV lookups, the system will attempt to connect to each address in priority order.

Connectivity error messages and using the support log

Diagnostic information is also recorded in the support log (Status > Support Log).

When Pexip Infinity connects successfully to the LDAP server, the support log will contain an entry similar to this:

```
2015-06-05T11:15:00.550+00:00 mgmt 2015-06-05 11:15:00,550 Level="INFO" Name="support.ldap" Message="Successfully connected to LDAP server" Address="server.example.com" Uri="ldaps://server.example.com"
```

Unable to contact the LDAP server

If Pexip Infinity cannot contact the configured LDAP server, the support log will contain an entry similar to this:

```
2015-06-05T08:40:29.707+00:00 mgmt 2015-06-05 08:40:29,704 Level="INFO" Name="support.ldap" Message="Failed connecting to LDAP server" Address="server.example.com" Reasons="ldaps://server.example.com : Can't contact LDAP server ldap://server.example.com : Can't contact LDAP server"
```

Ensure that the server is available at the configured address and, if the server address is specified by domain name or FQDN, ensure that DNS records exist and resolve to the correct address.

Connection errors: TLS certificate issues

If Pexip Infinity can reach the configured LDAP server, but cannot connect to it due to TLS certificate issues, the support log will contain an entry similar to this:

```
2015-06-05T08:55:49.042+00:00 mgmt 2015-06-05 08:55:49,042 Level="INFO" Name="support.ldap" Message="Failed connecting to LDAP server" Address="server.example.com" Reasons="ldaps://server.example.com : Can't contact LDAP server ldap://server.example.com : Connect error"
```

The reason "Connect error" means that Pexip Infinity cannot verify the LDAP server's certificate.

Ensure that the LDAP server's TLS certificate (or the CA certificate that signed it, if it is not self-signed) is in the Pexip Infinity trust store (Platform > Trusted CA Certificates).

Connection errors: binding to the server fails e.g. invalid credentials

If Pexip Infinity can reach the configured LDAP server, but cannot connect to it due to binding errors, such as invalid credentials, the support log will contain an entry similar to this:

```
2015-06-05T09:11:03.765+00:00 mgmt 2015-06-05 09:11:03,765 Level="INFO" Name="support.ldap" Message="Failed connecting to LDAP server" Address="server.example.com" Reasons="ldaps://server.example.com : Invalid credentials ldap://server.example.com : Invalid credentials"
```

Ensure that you have entered the correct credentials. They should be for an enabled, non-expired, domain user service account (not the Administrator account), which has a password set to never expire. All usernames and passwords are case sensitive.

If you are certain that the account you are trying to bind with is configured correctly, try to bind using the:

- bare username of the service account (e.g. `ldapuser`)
- full DN of the service account (e.g. `CN=ldapuser,CN=Users,DC=example,DC=com`)
- Windows logon of the service account (e.g. `EXAMPLE\ldapuser`).

Connection errors when using insecure transport

You cannot specify the LDAP server address as an IP address if you have also selected the **Allow insecure transport** option. If the server address is not specified as an FQDN you will receive "Invalid credentials" error messages.

You cannot use an IP address because the authentication handshake is encrypted using SASL technology. To achieve this, various shared keys are used — things both sides know and use as part of the handshake but are not exchanged on the wire. In this case, it is the FQDN of the LDAP server that is used.

Therefore, if you need to use insecure transport, you must ensure that you refer to the LDAP server by its FQDN (and this is the hostname the server uses to identify itself, not just something that points to the IP address), so that the authentication will work. See [Using ldapsearch or AD Explorer to view the LDAP database](#) below for an example of how to discover an AD server's hostname.

Alternatively, you could use secure transport, referring to the LDAP server by any name that appears in its TLS certificate, and by loading all necessary trusted CA certificates onto Pexip Infinity.

Connection errors: Error syncing with LDAP

You can receive an "Error syncing with LDAP" error message when attempting to perform a template synchronization when provisioning VMRs, devices or users.

This can be caused by invalid syntax in the template's **LDAP user filter** or **LDAP user search DN** fields. Check that all parentheses are balanced and are in the correct places, and that all operators are correctly positioned.

This message can also be received if you have not selected an LDAP sync source when configuring your sync template.

Cannot log in to Pexip Infinity despite using correct credentials

If users receive a "Please enter the correct username and password for an administrator account" message when trying to log in to Pexip Infinity, but they are using the correct username and password, this typically means that either:

- The LDAP server cannot be contacted:
 - These errors will be recorded in the Support log; see the connectivity troubleshooting guidelines above for more information.
- The LDAP server can be contacted but the correct user records are not being searched:
 - Check the Pexip Infinity LDAP configuration settings (**Users & Devices > User Authentication**) to ensure that all objectClass and LDAP field names have been spelled correctly, and that the base DN and user search DN fields contain the correct domain and organizational unit settings.
 - If you are using nested AD security groups, see [Supporting nested security groups in Windows Active Directory](#).
- The LDAP server can be contacted and the user records can be found and authenticated, but the user is not authorized to access Pexip Infinity:
 - Check that LDAP roles have been configured on Pexip Infinity (**Users & Devices > LDAP Roles**).
 - Ensure that the user's LDAP account is associated with the LDAP group DN's / role combinations that are configured on Pexip Infinity.

Recovering local access

If necessary you can reinstate access via the Pexip Infinity local on-box database, so that administrators can log in via the default account (typically **admin**) and will have full administrator privileges. You may need to do this if, for example, the **Authentication source** is configured as **LDAP database** and your connectivity to the LDAP server goes down or your credentials become invalid.

To reactivate your local admin account:

1. Log in to the Management Node over SSH.
2. For local admin access only, run the command:

```
authset LDAP LOCAL
```


or, for LDAP and local admin access, run the command:

```
authset LDAP BOTH
```

You can also disable client certificate authentication so that you can log in to the Pexip Infinity Administrator interface via the standard login page.

To disable certificate-based authentication:

1. Log in to the Management Node over SSH.
2. Run the command:

```
authset CBA OFF
```

If you forget the password for the Pexip Infinity Administrator interface, you can [re-run the installation wizard](#), being sure to change only the **Web administration password** setting.

VMRs, devices or user records not created as expected by a sync template

User search or user filters not being applied

If more or fewer VMRs, devices or users than expected (or no VMRs/devices/users at all) were created after performing a template synchronization, it is likely that the **LDAP base DN**, **LDAP user search DN** and **LDAP user filter** fields have been misconfigured.

Check that all objectCategory, objectClass and LDAP field names have been spelled correctly. Note that all LDAP user search and user filter contents are not case sensitive.

More information on Active Directory LDAP filtering can be found at

<https://social.technet.microsoft.com/wiki/contents/articles/5392.active-directory-ldap-syntax-filters.aspx>.

Using ldapsearch or AD Explorer to view the LDAP database

Mac and Linux systems

You can use a command line tool such as **ldapsearch**, which is available for Mac and Linux systems, to help test and diagnose connectivity issues with the LDAP server. Note that **ldapsearch** is installed by default on all Pexip Infinity nodes.

Here are some example **ldapsearch** queries you could use (after adapting the parameters as appropriate for your environment).

```
$ ldapsearch -v -h 10.0.0.8 -D "example\admin123" -w password123 -b OU=people,DC=example,DC=com
```

This fetches the contents of OU (org unit) **people** from the LDAP server at **10.0.0.8** over TCP, binding as user (sAMAccountName) **admin123** in NetBIOS domain **example** with password **password123** using simple (insecure) authentication.

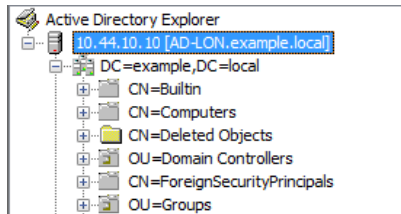
```
$ ldapsearch -v -h dc01.example.com -Y DIGEST-MD5 -U admin123 -w password123 -b OU=people,DC=example,DC=com
```

This extends the previous example by addressing the LDAP server by its FQDN **dc01.example.com** and uses SASL/DIGEST-MD5 authentication.

Windows

Windows users can use Active Directory Explorer (AdExplorer) to navigate around and view AD structures and entries. See <https://technet.microsoft.com/en-us/sysinternals/adexplorer> for more information and links to download the software.

The example below shows how you can discover your AD server's actual hostname (**AD-LON.example.local** in this case) if you use AdExplorer to connect to your server via its IP address (**10.44.10.10** in this case):



Using AD FS for client authentication

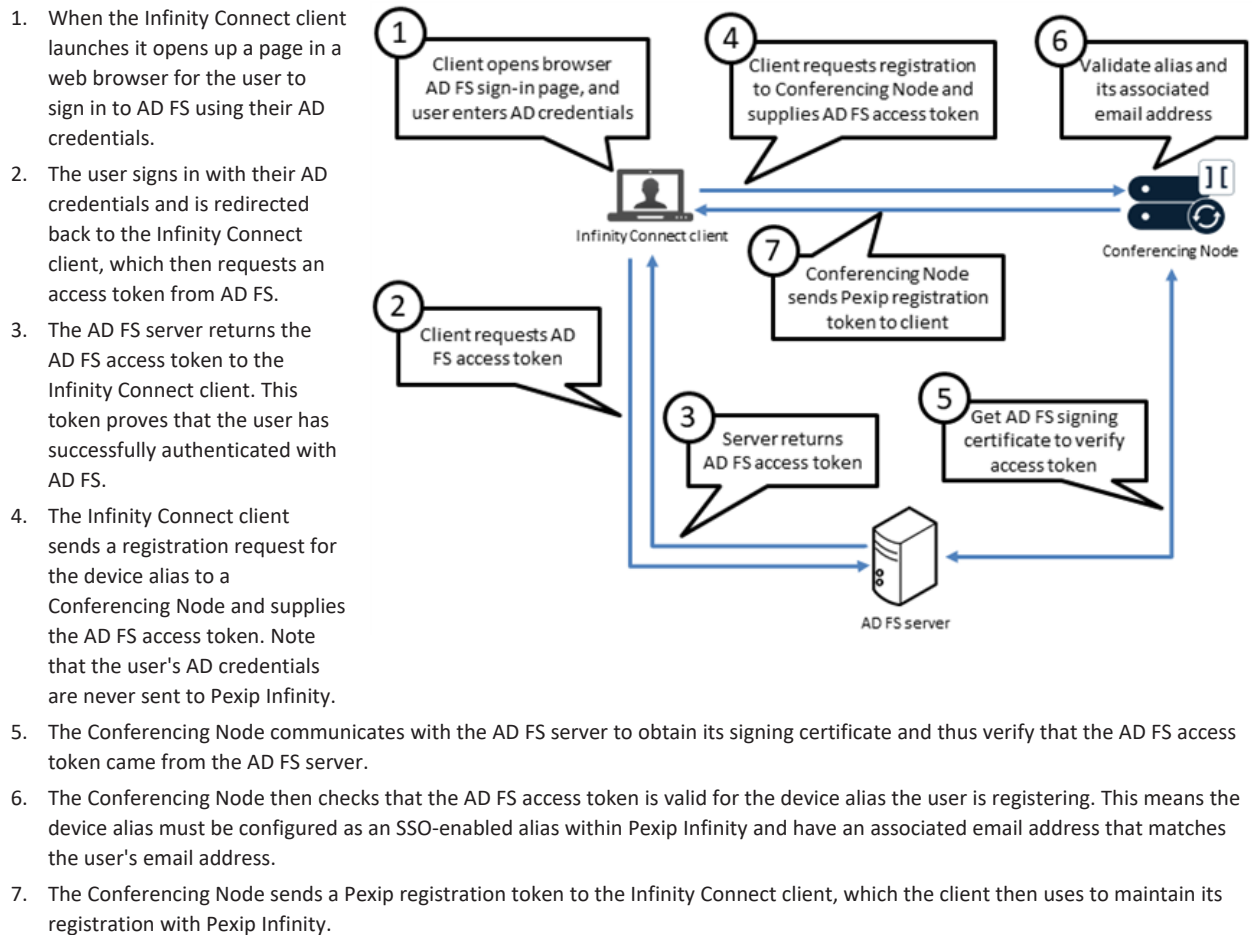
Pexip Infinity can integrate with Active Directory Federation Services (AD FS) to provide Infinity Connect clients and other third-party applications with single sign-on access. This allows users to register their clients using their AD credentials.

This topic explains how to configure AD FS and Pexip Infinity to enable users to register their clients using their AD FS credentials. It covers:

- [How it works](#)
- [Prerequisites](#)
- [Setting up an OAuth 2.0 Client on Windows Server](#)
- [Registering and provisioning Infinity Connect](#)
- [Troubleshooting](#)

How it works

The process of authenticating and registering an Infinity Connect client to Pexip Infinity via end-user SSO with AD FS works as follows:



Note that:

- The AD FS access token lasts for 8 hours. The Infinity Connect client automatically opens the sign-in page when it needs a new AD FS access token. This typically occurs when the user loads the client for the first time in the day. However, if the user is already signed into AD FS, then they might not notice anything because they will be immediately redirected back to the client without needing to sign in.
- A Conferencing Node requests the signing certificate from the AD FS server the first time it needs to validate a token and then caches it for one hour for subsequent SSO registration requests.
- The Pexip registration token is used by the Infinity Connect client to periodically refresh its registration with the Conferencing Node. This means that while the client remains registered, it does not matter if the AD FS access token expires. But if the client becomes unregistered (e.g. due to a long network connection failure) and the AD FS token has expired, then the user is asked to sign in to AD FS again.
- The legacy Infinity Connect clients do not support AD FS authentication.

Prerequisites

Before you integrate your AD FS deployment with Pexip Infinity, you must make sure your AD FS deployment satisfies the following requirements.

AD FS version

You must be using a version of Windows Server that supports OAuth 2.0 with AD FS, i.e. Windows Server 2012 R2 or later.

Internet accessibility and security

Your Federation Service must be accessible by:

- all Pexip Infinity Conferencing Nodes
- all users who need to sign into AD FS to authenticate with Pexip Infinity.

In practice this means your Federation Service must be accessible from the internet. This raises security concerns, but Microsoft provide documentation about the recommended deployment of AD FS:

- Top-level AD FS documentation can be found at <https://docs.microsoft.com/en-us/windows-server/identity/active-directory-federation-services>.
- Note the network layout recommendations made in the Microsoft documentation, for example <https://docs.microsoft.com/en-us/windows-server/identity/ad-fs/design/federation-server-farm-using-sql-server> where all AD FS servers are deployed inside the corporate network and are load-balanced.
- To make the Federation Service accessible from the internet, separate servers in the DMZ can be installed with the Web Application Proxy (WAP) role, which proxy requests to the internal AD FS servers from the internet.
- Your Federation Service Name, e.g. adfs.example.com, must be routable from both inside and outside your corporate network.
 - Inside your corporate network, it should resolve directly to your AD FS server (or the IP address used to load balance multiple AD FS servers).
 - Outside your corporate network, it should resolve to your WAP servers in the DMZ.

This requires the correct DNS configuration to be setup. Microsoft also provide documentation about this, for example <https://docs.microsoft.com/en-us/windows-server/identity/ad-fs/deployment/configure-name-resolution-for-a-federation-server-proxy-in-a-dns-zone-that-serves-only-the-perimeter-network>.

- Another very good source of information for creating a highly available AD FS deployment can be found in the series of blog posts at <https://blogs.technet.microsoft.com/platformspfe/2014/08/28/part-1-windows-server-2012-r2-ad-fs-federated-web-sso/>.

These posts refer to AD FS 3.0 on Windows Server 2012 R2, but they also apply to AD FS 4.0 on Windows Server 2016.

AD account with email address for each user

Pexip Infinity uses email addresses to identify users. This means every user in your organization who needs to authenticate to Pexip Infinity must have an Active Directory account that includes an email address.

Certificates

Each AD FS server must be provided with a valid certificate which is trusted by your Pexip Infinity deployment. The subject of this certificate needs to match the Federation Service Name.

Setting up an OAuth 2.0 Client on Windows Server

To set up an OAuth 2.0 Client, use the appropriate set of instructions below for your version of AD FS and Windows Server.

- Windows Server 2016 and later
- AD FS 3.0 on Windows Server 2012 R2

Creating a Native Application

In this step you create a new application group with a new native application for the OAuth client — which is the Infinity Connect client in this case.

1. Log on to a computer that can make configuration changes to your Federation Service. If your AD FS deployment uses Windows Internal Database (WID), this must be the Primary AD FS Server. If your AD FS deployment uses SQL Server then any AD FS server can make configuration changes.
2. From the top right of the Server Manager application window, select **Tools > AD FS Management**.
3. From the left panel, select **Application Groups** and then from the right panel select **Add Application Group**.
4. At the **Welcome** screen, enter a **Name** and **Description** for the application group. Then from the **Template** section, under **Standalone applications**, select **Native application**.

- At the **Native Application** screen, enter a **Name** for the application. A new **Client Identifier** is randomly generated for you (this will be required later when [adding the OAuth client details](#) to the Management Node).


In the **Redirect URI** field, enter:

https://<address>/api/client/v2/oauth2_redirect

where <address> is the FQDN of a Conferencing Node or reverse proxy, for example

https://px01.vc.example.com/api/client/v2/oauth2_redirect

You may want to use a reverse proxy rather than a Conferencing Node for the redirect URI if you want to provide some redundancy capability.

-  The **Redirect URI** you enter here must match exactly the URI used when provisioning the Infinity Connect client.

- At the **Summary** screen, review the settings and select **Next**.

The new application group, along with the new native application, is created.

Creating a Web API Resource

In this step you create a Web API within your application group. The Web API acts as the resource that is accessed when users authenticate to Pexip Infinity using their AD credentials.

- On the AD FS Management Tool, from the left-hand panel select **Application Groups** and from the middle panel select the application group you have just created. From the right-hand panel, select **Properties**.
- At the bottom of the **Properties** window, select **Add application...**
- At the **Welcome** screen, from the **Template** section, select **Web API**.
- At the **Configure Web API** screen, enter a **Name** and an **Identifier** (which must be in URL format).
- At the **Choose Access Control Policy** screen select an appropriate policy. The default is *Permit everyone*.
- At the **Configure Application Permissions** screen, ensure the native application you [created above](#) appears in the list of **Client applications**. All the **Permitted scopes** should be deselected because the Pexip Infinity apps do not request any scopes.
- At the **Summary** screen, review the entered settings then select **Next**.

The new Web API should now be created.

Configuring Claim Rules

In this step you configure an Issuance Transform Rule for the Web API. This rule specifies which claims should be sent to the Relying Party (i.e. which claims will be inside the OAuth token that is sent to Pexip Infinity).


Pexip Infinity requires certain claims to be present inside the token in order to establish the user's identity. These are claims that come from the user's Active Directory account.

- Ensure there is an Attribute Store configured for Active Directory. To do this, from the AD FS Management Tool, in the left-hand panel expand **Service** and select **Attribute Stores**. Select the attribute store called Active Directory. Open its properties and ensure its **Attribute store type** is *Active Directory*.

If the Attribute Store is not present, add it by selecting **Add Attribute Store**. In the **Add An Attribute Store** window, enter a **Display name** of *Active Directory* and select an **Attribute store type** of *Active Directory*.

- Add the Issuance Transform Rule on the Web API you just created. To do this, from the AD FS Management Tool, at the bottom of the left-hand panel select **Application Groups**. Select the Application Group you just created and open its **Properties**.
- Select the [Web API you created earlier](#) and then select **Edit**.
- Select the **Issuance Transform Rules** tab and then select **Add Rule**.
- Select a **Claim rule template** of *Send Claims Using a Custom Rule* and then select **Next**.
- Enter a **Claim rule name** and enter the following as the **Custom rule**.

```
c:[Type ==
"http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname", Issuer == "AD AUTHORITY"]
=> issue(store = "Active Directory", types = ("email", "object_guid", "first_name", "last_name", "display_name"), query =
";mail,objectGUID,givenName,sn,displayName;()", param = c.Value);
```

-  The above rule queries Active Directory for the attributes: **mail**, **objectGUID**, **givenName**, **sn** and **displayName**, and then maps them to the claims: **email**, **object_guid**, **first_name**, **last_name** and **display_name** which will appear in the token payload that is returned when the user successfully logs in. The **email** and **object_guid** claims are required by Pexip Infinity when verifying the token. If they are not present in the token, the user will fail to authenticate to Pexip Infinity.

- Select **Finish**, and in the next window ensure you select **Apply**.

Checking and enabling AD FS endpoints

In this step you ensure that the appropriate AD FS endpoints have been enabled to support Pexip's requirements. In the context of AD FS, an endpoint is a URL that AD FS is configured to serve.

To find the details of these AD FS endpoints:

1. From the Server Manager application window, select **Tools > AD FS Management**.
2. From the AD FS Management Tool, in the left-hand panel expand **AD FS > Service > Endpoints**.
3. Locate and check the following 2 endpoints:
 - an **OAuth** type endpoint with path **/adfs/oauth2/**
(this is used by users who sign in to AD FS)
 - a **Federation Metadata** type endpoint with path **/FederationMetadata/2007-06/FederationMetadata.xml**
(this is used by Conferencing Nodes)

Ensure that both of these endpoints are **Enabled**. If you are using a Web Application Proxy (WAP) you must also ensure they are **Proxy Enabled**.

Determining Federation Service Properties

To add an AD FS OAuth 2.0 Client to the Pexip Infinity Management Node, you must first determine your **Federation Service Name** (this is the FQDN that clients use to access AD FS) and **Federation Service Identifier**. To check these:

- From the AD FS Management Tool, from the left-hand panel select the top level **AD FS** folder, and then select **Edit Federation Service Properties**.

In the example below, the **Federation Service Name** is **adfs.rd.pexip.com** and the **Federation Service identifier** is **http://adfs.rd.pexip.com/adfs/services/trust**.

Adding the AD FS OAuth 2.0 Client to the Pexip Infinity Management Node

1. From the Pexip Infinity Management Node, go to **Users & Devices > AD FS Authentication Clients**.
2. Select **Add AD FS OAuth 2.0 Client**.
3. Complete the fields as follows:

Field	Description
Name	The name to use to refer to this OAuth 2.0 client on AD FS.
Description	An optional description of this OAuth 2.0 client.
AD FS Server configuration	
Federation Service name	The FQDN which is used by clients to connect to AD FS. This is the name seen when Determining Federation Service Properties .
Federation Service Identifier	The URL which identifies AD FS. This is the identifier seen when Determining Federation Service Properties .
Resource Identifier	The URL which identifies the OAuth 2.0 resource on AD FS. This should be the identifier you gave the Web API resource which you created earlier.
Client ID	The ID of the AD FS OAuth 2.0 Client. This will show a randomly generated GUID but you must replace this with the one that was generated earlier when Creating a Native Application .
AD FS Domains	
AD FS Domain: #1	

Field	Description
Domain	<p>An FQDN which, when present in a user's email address, permits that user to authenticate using this AD FS OAuth 2.0 client.</p> <p>For example, if a user has the email bob@example.com, then example.com must be configured as a Domain to allow that user to sign on with AD FS.</p> <p>A single domain can only be used by a single AD FS 2.0 OAuth client.</p>
Description	An optional description of the AD FS domain.

4. Select **Save and Test Connection**. This will save your data and report success or failure details of this connection. See [Troubleshooting](#) for more information.

Creating a Relying Party Trust

In this step you create a Relying Party Trust, which acts as the resource that is accessed when users authenticate to Pexip Infinity using their AD credentials.

1. Log on to a computer that can make configuration changes to your Federation Service. If your AD FS deployment uses Windows Internal Database (WID), this must be the Primary AD FS Server. If your AD FS deployment uses SQL Server then any AD FS server can make configuration changes.
2. From the Server Manager application window, select **Tools > AD FS Management**.
3. From the left-hand panel, expand **AD FS > Trust Relationships > Relying Party Trusts**. Then from the right-hand panel select **Add Relying Party Trust....**
4. At the **Add Relying Party Trust Wizard** welcome screen, select **Start**.
5. At the **Select Data Source** screen, select **Enter data about the relying party manually** and select **Next**.
6. At the **Specify Display Name** screen, enter a **Display name** and **Note**, and select **Next**.
7. At the **Choose Profile** screen, select **AD FS profile** and select **Next**.
8. At the **Configure Certificate** screen, do not configure a certificate. Simply select **Next**.
9. At the **Configure URL** screen, do not enable support for either the WS-Federation Passive or the SAML 2.0 WebSO protocols. Simply select **Next**.
10. At the **Configure Identifiers** screen, enter a Relying Party Trust Identifier. This must be unique against all of your Relying Party Trusts, and must be in URL format.
11. At the **Configure Multi-factor Authentication Now?** screen, optionally enable multi-factor authentication. Enabling multi-factor authentication will affect how your users sign in to AD FS.
12. At the **Choose Issuance Authorization Rules** screen, select **Permit all users to access this relying party**.
13. At the **Ready To Add Trust** screen, review the settings and then select **Next**.
14. At the **Finish** screen, verify that the relying party trust was added successfully. Choose to **Open the Edit Claim Rules dialog...** then select **Close**.

Configuring Claim Rules


In this step you configure an Issuance Transform Rule for the Relying Party. This rule specifies which claims should be sent to the Relying Party (i.e. which claims will be inside the OAuth token that is sent to Pexip Infinity).

Pexip Infinity requires certain claims to be present inside the token in order to establish the user's identity. These are claims that come from the user's Active Directory account.

1. Ensure there is an Attribute Store configured for Active Directory. To do this, in the AD FS Management Tool, from the left-hand panel expand **AD FS > Trust Relationships > Attribute Stores**. Look for the attribute store called **Active Directory**. Open its properties and ensure its **Attribute store type** is **Active Directory**.
If the Attribute Store is not present, add it by selecting **Add Attribute Store**. In the **Add An Attribute Store** window, enter a **Display name** of **Active Directory** and select an **Attribute store type** of **Active Directory**.
2. From the left-hand panel expand **AD FS > Trust Relationships > Relying Party Trusts**, select the Relying Party you just created, and from the right-hand panel select **Edit Claim Rules....**
3. Select the **Issuance Transform Rules** tab, then select **Add Rule....**

4. At the **Select Rule Template** screen, select a **Claim rule template** of *Send Claims Using a Custom Rule*.
5. At the **Configure Rule** screen, enter a **Claim rule name**, and in the **Custom rule** section enter the following:

```
c:[Type ==
"http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname", Issuer == "AD AUTHORITY"]
=> issue(store = "Active Directory", types = ("email", "object_guid", "first_name", "last_name", "display_name"), query =
";mail,objectGUID,givenName,sn,displayName;{0}", param = c.Value);
```

-  The above rule queries Active Directory for the attributes: **mail**, **objectGUID**, **givenName**, **sn** and **displayName**, and then maps them to the claims: **email**, **object_guid**, **first_name**, **last_name** and **display_name** which will appear in the token payload that is returned when the user successfully logs in. The **email** and **object_guid** claims are required by Pexip Infinity when verifying the token. If they are not present in the token, the user will fail to authenticate to Pexip Infinity.

6. Select **Finish**.
7. You are returned to the **Issuance Transform Rules** tab. Select **Apply**.

Checking and enabling AD FS endpoints

In this step you ensure that the appropriate AD FS endpoints have been enabled to support Pexip's requirements. In the context of AD FS, an endpoint is a URL that AD FS is configured to serve.

To find the details of these AD FS endpoints:

1. From the Server Manager application window, select **Tools > AD FS Management**.
2. From the AD FS Management Tool, in the left-hand panel expand **AD FS > Service > Endpoints**.
3. Locate and check the following 2 endpoints:
 - an **OAuth** type endpoint with path **/adfs/oauth2/**
(this is used by users who sign in to AD FS)
 - a **Federation Metadata** type endpoint with path **/FederationMetadata/2007-06/FederationMetadata.xml**
(this is used by Conferencing Nodes)

Ensure that both of these endpoints are **Enabled**. If you are using a Web Application Proxy (WAP) you must also ensure they are **Proxy Enabled**.

Determining Federation Service Properties

To add an AD FS OAuth 2.0 Client to the Pexip Infinity Management Node, you must first determine your **Federation Service Name** (this is the FQDN that clients use to access AD FS) and **Federation Service Identifier**. To check these:

- From the AD FS Management Tool, from the left-hand panel select the top level **AD FS** folder, and then select **Edit Federation Service Properties**.

In the example below, the **Federation Service Name** is **adfs.rd.pexip.com** and the **Federation Service identifier** is **http://adfs.rd.pexip.com/adfs/services/trust**.

Adding the AD FS OAuth 2.0 Client to the Pexip Infinity Management Node

In this step you add the details of the OAuth 2.0 client to the Pexip Infinity Management Node.

1. From the Pexip Infinity Management Node go to **Users & Devices > AD FS Authentication Clients**.
2. Select **Add AD FS OAuth 2.0 Client**.

- Complete the fields as follows:

Field	Description
Name	The name to use to refer to this OAuth 2.0 client on AD FS.
Description	An optional description of this OAuth 2.0 client.
AD FS Server configuration	
Federation Service name	The FQDN which is used by clients to connect to AD FS. This is the name seen when Determining Federation Service Properties .
Federation Service Identifier	The URL which identifies AD FS. This is the identifier seen when Determining Federation Service Properties .
Resource Identifier	The URL which identifies the OAuth 2.0 resource on AD FS. This should be the identifier you gave the Relying Party Trust which you created earlier.
Client ID	The ID of the AD FS OAuth 2.0 Client. This will show a randomly generated GUID which you may leave as is.
AD FS Domains	
AD FS Domain: #1	
Domain	An FQDN which, when present in a user's email address, permits that user to authenticate using this AD FS OAuth 2.0 client. For example, if a user has the email <code>bob@example.com</code> , then <code>example.com</code> must be configured as a Domain to allow that user to sign on with AD FS. A single domain can only be used by a single AD FS 2.0 OAuth client.
Description	An optional description of the AD FS domain.

- Select **Save**.

Adding the OAuth 2.0 Client to AD FS

In this final step you add the OAuth 2.0 client to AD FS. This is done using the `Add-AdfsClient` PowerShell command.

This command can be generated for you on the Pexip Infinity Management Node as follows:

- From the Pexip Infinity Management Node, select the AD FS OAuth 2.0 client you just created.
- From the bottom of the page, select **Save and Get Setup Config**.
- On this page, copy the command in the **Add-AdfsClient** field.

The command will look something like:

```
Add-AdfsClient -Name "AD FS 2012" -ClientId "c5bddcf3-f3fd-48c0-acc4-6d1af4ddf434" -RedirectUri @("pexip-auth://adfs",
"https://<Conference Node or Reverse Proxy>/api/client/v2/oauth2_redirect") -Description "AD FS Server for Connect clients"
```

This uses the **Name**, **Description** and **Client ID** configured for this AD FS OAuth 2.0 Client.

It also specifies two redirect URIs that may be used when provisioning the Infinity Connect clients. In the second URI you must replace `<Conference Node or Reverse Proxy>` with the actual address of either your Conferencing Node or your reverse proxy.

Note that the `pexip-auth://adfs` is an alternative redirect URI that immediately redirects the user back to the app. This redirect method causes the AD FS sign-in page to remain open and thus may cause user confusion as it is not clear the user has successfully signed in. You can enter both types of redirect URI when configuring AD FS. The redirect URI that is actually used is the one that the Infinity Connect client is provisioned with.

 One of the Redirect URIs you enter here must match exactly the URI used when provisioning the Infinity Connect client.

- On your AD FS server, open PowerShell and issue the above command.
- You must also enable Refresh tokens for the Relying Party Trust. This is done using the following PowerShell command:

```
Set-AdfsRelyingPartyTrust -TargetName "<Relying Party Trust Display Name>" -IssueOAuthRefreshTokensTo AllDevices
```

Where `<Relying Party Trust Display Name>` is the **Display name** of the **Relying Party Trust** which was created earlier.

- You can now test your connection. Return to the Pexip Infinity Management Node, select the AD FS client again and select **Save and Test Connection**. This will save your data and report success or failure details of this connection. See [Troubleshooting](#) for more information.

Other commands are also available from the **AD FS Setup Config** page:

- To verify the changes have been saved, use the `Get-AdfsClient` command.
- If you need to modify the client, use the `Set-AdfsClient` command.

For information about these and other relevant commands, see the following Microsoft documentation:

- Add-AdfsClient: <https://docs.microsoft.com/en-us/powershell/module/adfs/add-adfsclient>
- Set-AdfsClient: <https://docs.microsoft.com/en-us/powershell/module/adfs/set-adfsclient>
- Get-AdfsClient: <https://docs.microsoft.com/en-us/powershell/module/adfs/get-adfsclient>
- Enable-AdfsClient: <https://docs.microsoft.com/en-us/powershell/module/adfs/enable-adfsclient>
- Disable-AdfsClient: <https://docs.microsoft.com/en-us/powershell/module/adfs/disable-adfsclient>
- Set-AdfsRelyingPartyTrust: <https://docs.microsoft.com/en-us/powershell/module/adfs/set-adfsrelyingpartytrust>

Users should now be able to use AD FS services and their AD credentials to register their Infinity Connect clients to Pexip Infinity.

Registering and provisioning Infinity Connect

When your AD FS integration is complete, you can provision your Infinity Connect users with the relevant settings so that they can use AD FS services and their AD credentials to register their Infinity Connect clients to Pexip Infinity.

See [Registering and provisioning the Infinity Connect client](#) for instructions about how to do this and for details of the associated end-user experience.

Troubleshooting

You can test your AD FS configuration by going to **Users & Devices > AD FS Authentication Clients**, selecting the client you want to test, and then from the bottom of the page selecting **Save and Test Connection**.

The page will refresh and display one or more diagnostic messages indicating success or failure. Example error messages are:

Error message	Possible cause and resolution
Unable to connect to the Federation Metadata located at <code>https://<address>/FederationMetadata/2007-06/FederationMetadata.xml</code> .	<p>Check that the Federation Service Name FQDN is correct and reachable.</p> <p>Each AD FS server must be provided with a valid certificate which is trusted by your Pexip Infinity deployment. The subject of this certificate needs to match the Federation Service Name.</p>
The Entity ID 'http://<address>/adfs/services/trust' was found in the Federation Metadata located at <code>https://<address>/FederationMetadata/2007-06/FederationMetadata.xml</code> . This differs from the AD FS Identifier entered below. Please make sure it is entered correctly.	There is a discrepancy between the Federation Service Identifier configured in Pexip Infinity and what is configured in the AD Federation Service Properties.
No signing certificates could be found in the Federation Metadata located at <code>https://<address>/FederationMetadata/2007-06/FederationMetadata.xml</code> .	Each AD FS server must be configured with at least one Token-signing certificate. You can check these in your AD FS configuration by going to Service > Certificates and making sure at least one Token-signing certificate is listed and has not expired.

Typical success messages (no action is required) include:

- "Successfully found one signing certificate in the Federation Metadata located at `https://<address>/FederationMetadata/2007-06/FederationMetadata.xml`."
- "Successfully verified AD FS Identifier matches the Entity ID 'http://<address>/adfs/services/trust' in the Federation Metadata located at `https://<address>/FederationMetadata/2007-06/FederationMetadata.xml`."

Pexip Infinity maintenance tasks

Typical maintenance tasks include upgrading the Pexip Infinity software, creating and restoring backups of configuration data, and managing administrator account passwords.

Backing up and restoring configuration

You should take regular backups of the configuration data on your Management Node, particularly before and after [Upgrading the Pexip Infinity platform](#). This allows you to restore your configuration to a specific point in time, or to restore your configuration if you have to deploy a new Management Node.

There are two ways to maintain copies of your Management Node configuration data:

- use your [hypervisor's tools](#) to create a full backup or snapshot of the Management Node VM
- use the [backup and restore mechanism](#) built into the Pexip Infinity Administrator interface (automatic and manual backup options are available)

For on-premises deployments, we recommend that you use both the hypervisor and Pexip's inbuilt methods to preserve your configuration data. A VM snapshot should be your primary mechanism prior to an upgrade, as this allows you to easily restore your system back to its state at the time the snapshot was taken. The Pexip Infinity backup and restore mechanism is your fallback mechanism, as this allows you to preserve a copy of your data in an alternative location, in case you lose your VM environment. Cloud-based deployments (Azure, AWS, GCP or Oracle) should use the Pexip Infinity backup and restore mechanism.

You can also separately backup and restore just your Virtual Meeting Room and Virtual Auditorium configuration, see [Bulk import/export of service configuration data](#).

If you are using VMR Scheduling for Exchange, we recommend that you run the [scheduling recovery script](#) after restoring the Management Node to ensure that any meetings that were scheduled after the backup was taken are reinstated.

See [Resilience strategies — redundancy, backing up and restoring data](#) for more general guidance on resilience strategies.

Backing up Conferencing Nodes

Conferencing Nodes do not need to be backed up. They receive all their configuration information from the Management Node and can simply be redeployed if necessary. However, if your Conferencing Nodes are geographically spread out and redeploying them would consume significant bandwidth or take a significant length of time, they can also be backed up with your hypervisor's backup tools.

Using your hypervisor's tools to take and restore backups

You can use your hypervisor's own backup tools, or any other third-party tool that supports VM backups, to create a full backup of the Management Node. This backup can then be re-deployed at a later date if required.

Be aware that snapshots are not backups. Snapshots are a tool to roll back to a given time. Therefore, we recommend taking snapshots only when necessary (such as prior to an upgrade) and deleting the snapshot as soon as possible after the upgrade is confirmed to be successful. You should only create and delete VMware snapshots at a time of minimal usage. Taking or removing snapshots can cause virtual machines to become unresponsive.

VM backups should use a proper hypervisor VM backup tool (e.g. VMware VDP — vSphere Data Protection) or similar, and restoration should be tested and verified (preferably after the inbuilt backup methods have been set up, to ensure that you have another way of recovering if your restoration fails).

Consult your hypervisor documentation for more information about taking and restoring backups and snapshots.

Using backup and restore via the Pexip Infinity Administrator interface

You can use Pexip Infinity's inbuilt backup and restore mechanism to backup and restore the configuration data on the Management Node.

You can enable daily automatic backups, and you can also take a manual backup whenever it is appropriate, for example, before and after you make any configuration changes or perform a software upgrade.

- All backup files are encrypted — the administrator supplies a passphrase and must remember this for any subsequent restoration.
- Restoration must occur on exactly the same version that the backup was taken from.
- The data contained in the backup contains all configuration data, including IP addresses, custom themes, certificates and call history.
- The backup data does not contain licenses, the administrator log, the support log, usage statistics or the operating system password.
- The system keeps on the host VM only the 5 most recent manually-taken backups and the 5 most recent automatic backups. Older backup files are deleted.

Note that this function can only be used to restore configuration or to replicate the configuration of a previous Management Node onto a new Management Node. It cannot be used to redeploy Conferencing Nodes.

Managing backup files

The system keeps on the host VM only the 5 most recent manually-taken backups and the 5 most recent automatic backups. Older backup files are deleted.

- To see the backup files that are currently stored on the host VM, go to **Utilities > Backup/Restore** and look at the list of **Existing Backup Files**.
- If you want to manually download a backup file from the host VM to another machine, go to **Utilities > Backup/Restore** and select the **Download backup** option. You can also configure the daily automatic backup to upload each backup file to an external FTP server.

Enabling daily automatic backups

You can enable Pexip Infinity to automatically backup the Management Node configuration data on a daily basis.

When automatic backups are enabled:

- Each backup is taken at 01:02 UTC every day.
- Successful backup operations are recorded in the administrator log (with a "Created automatic system backup" message).
- Automatic backup filenames take the format:
`pexip_auto_backup_<hostname>_<version>_<build>_<date>_<time>.tar.pexbak`

To enable automatic backups:

1. Go to **Utilities > Automatic Backups**.
2. Select the **Enable automatic backups** checkbox (this is disabled by default).
3. Enter a **Backup passphrase**.
The text entered here is used to encrypt the backup file. You must remember this text as it will be required if you need to subsequently restore the data from the file.
4. The system always keeps the 5 most recent automatic backups on the host VM. In addition, you can upload each automatic backup file to an external FTP server. To do this, you must specify the **Upload URL** (supported schemes are FTPS and FTP) and the **Username** and **Password** credentials of the FTP server.
5. Select **Save**.

Manually creating a backup file

To manually create and download a backup file:

1. Go to **Utilities > Backup/Restore**.
2. In the **Create backup** section, enter a **Passphrase** and then enter it again in the **Re-enter passphrase** field.
The text entered here is used to encrypt the backup file. You must remember this text as it will be required if you need to subsequently restore the data from the file.
3. Select **Create backup**.
After a few seconds you see a message: "Successfully created the backup file: <file_name>" where <file_name> takes the format:
`pexip_backup_<hostname>_<version>_<build>_<date>_<time>.tar.pexbak`

4. Download the file from the host VM:
 - a. From the **Existing backup files** section at the bottom of the page, select **Download backup** for the file you have just created.
 - b. Follow your browser's prompts to save or download the file to your local file system.
 - c. If required, you can delete unwanted backup files from your host VM by selecting **Delete backup**.

The system keeps on the host VM only the 5 most recent manually-taken backups and the 5 most recent automatic backups. Older backup files are deleted.

Restoring data to the Management Node

You can restore configuration data to the Management Node. This could be restored to the original Management Node from which the backup was taken, or it could be restored to a newly deployed Management Node (if the original node was lost due to, for example, issues with its VM environment).

Any in-progress calls will not be affected while data is being restored to the Management Node.

To restore configuration data:

1. If required, deploy a new Management Node (see [Installation overview](#) for links to the appropriate instructions for your hypervisor):
 - Complete the installation wizard as normal.
 - The IP address you enter at this stage is temporary (it can be the same as the previous Management Node).
 - All the configuration data you enter, including the IP address, will be subsequently replaced.
2. If any new Conferencing Nodes have been deployed since the backup was taken:
 - Power these Conferencing Nodes off and delete them before restoring the Management Node data.
 - These additional Conferencing Nodes will not be recognized by the restored Management Node. You will have to create them again after the restore has completed.
3. If previously configured Conferencing Nodes have been deleted since the backup was taken:
 - There is no need to do anything at this stage; the restore will complete successfully.
 - Simply delete the Conferencing Nodes from the restored configuration, after the restore has completed.
4. Restore your previous backup file:
 - a. Go to **Utilities > Backup/Restore**.
 - b. In the **Restore backup** section, enter the **Passphrase**.

The text entered here must be identical to the text that was used to create the backup file.
 - c. Select **Choose File** and then choose the backup file that you want to restore.

The file must be chosen from your local file system (you cannot select a file from the list of **Existing backup files**).
 - d. Select **Restore backup**.

You are taken to the **Restore Backup** confirmation page.

If instead, you see "Failed to restore the system from the backup file" with a "Decryption Error: decryption failed" message, the most likely reason for this is that you have entered an incorrect passphrase.
 - e. The confirmation page shows the date that the backup was taken, and the Management Node IP address that will be restored to this system.

Select **Restore backup** to confirm the restoration.
5. If the restore is successful, after a few seconds you will see a "Successfully restored the backup file" message and the Management Node will reboot.

You need to wait for the node to return from rebooting before you can access it again.

If you have restored a file where the IP address of the Management Node in the backup file is different from the node's current address (prior to the restore), you need to manually enter this IP address into the web browser to access the restored Management Node's web interface login page.
6. If you have restored your configuration onto a new VM (and were unable to return the licenses from your previous VM), you must contact your Pexip authorized support representative to get your licenses reapplied on your new VM.

Upgrading the Pexip Infinity platform

- i** We have designed the upgrade process to minimize disruption. However, there may be some temporary loss of functionality or unpredictable system behavior due to Conferencing Nodes running conflicting software versions, or some Conferencing Nodes being placed in maintenance mode. For this reason, we recommend upgrading at a time of minimal usage.

The upgrade process

When you initiate an upgrade of the Pexip Infinity software, the following steps occur automatically:

1. The Management Node software is upgraded, after which the Management Node automatically reboots.
2. The first 10 Conferencing Nodes are put into [maintenance mode](#). This means that all further incoming calls to those nodes will be rejected.
(In Pexip Infinity deployments of 10 Conferencing Nodes or fewer, all of the nodes are placed into maintenance mode.)
3. When all calls have cleared from a Conferencing Node that is in maintenance mode, its software is upgraded and the node is rebooted. This process should take around 2 minutes to complete, but may take longer depending on the connection speed between the Management Node and the Conferencing Node (as the upgrade files have to be transferred between the nodes).
If all of the calls on a Conferencing Node that is in maintenance mode have not cleared after 1 hour, the node is taken out of maintenance mode and put at the back of the queue of nodes to be upgraded. A further attempt to upgrade that node will be made after all other nodes have been upgraded (or had upgrade attempts made).
4. After a Conferencing Node has been upgraded successfully (or has been put back in the queue for a later upgrade attempt) and is again available, another Conferencing Node is selected and put into maintenance mode.
5. The process continues with each subsequent Conferencing Node being put into maintenance mode, and, after all calls have cleared, being upgraded and then rebooted. Up to 10 Conferencing Nodes may simultaneously be in maintenance mode or in the process of being upgraded at any one time.
Any Conferencing Nodes used for dynamic cloud bursting will be automatically started up and upgraded. Bursting nodes are normally selected for upgrade after the system has upgraded (or attempted to upgrade) the "always-on" nodes.
6. If the upgrade of the Management Node and all Conferencing Nodes has not completed successfully after 24 hours, the process will stop and all nodes will be left in their existing upgrade state. This is designed to prevent situations where some Conferencing Nodes cannot be upgraded, which would otherwise leave the system in a permanent state of upgrading.

If the upgrade process does not complete successfully and stops after 24 hours, you may have a mix of upgraded and non-upgraded nodes. You will then need to repeat the upgrade process. During a repeat upgrade, only those nodes that have not already been upgraded will be included in the upgrade process.

During the 24-hour period from when an upgrade has been initiated, you cannot re-initiate an upgrade using the Administrator interface. If you must re-initiate an upgrade during this time, you must reboot the Management Node and then start the process again.

When to upgrade

While the upgrade is in progress, some Conferencing Nodes will be running the newer version of the software and some will be running the older version. These Conferencing Nodes will be incompatible until they are all again running the same version. This means that there may be instances where two endpoints dial the same Virtual Meeting Room alias but if they are routed to different Conferencing Nodes that are running different versions of the software, the two endpoints will be in different conferences. It may also mean that, if an endpoint is in an ongoing call on a node that is put into maintenance mode in preparation for an upgrade, some call functionality such as presentation sharing may be limited. For this reason, **we recommend upgrading at a time of minimal usage**.

Alternatively, to avoid unpredictable system behavior due to Conferencing Nodes running conflicting software versions, you may want to manually put **all** of your Conferencing Nodes into [maintenance mode](#) before initiating the upgrade process. This will allow all existing calls to finish, but will not admit **any** new calls. You should then actively monitor your Conferencing Nodes' status and manually take each node out of maintenance mode after it has been upgraded to the new software version, so that the system can start taking new calls again on those upgraded nodes.

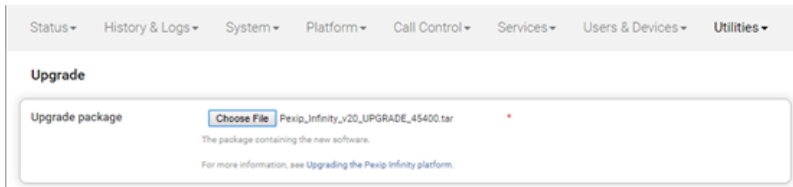
Upgrading from version 22 or later to version 26

To upgrade Pexip Infinity software from v22 or later to v26:

1. Before upgrading an on-premises deployment, we recommend that you use your hypervisor's snapshot functionality to take a full VMware/Hyper-V snapshot of the Management Node. You may also want to take a snapshot of each Conferencing Node, although depending on the size and complexity of your deployment it may be easier to simply redeploy these from the Management Node (after it has been rolled back) in the unlikely event that this is required.

Before upgrading a cloud-based deployment (Azure, AWS, GCP or Oracle), you should backup the Management Node via Pexip Infinity's inbuilt mechanism (**Utilities > Backup/Restore**).

2. Download the [Pexip Infinity upgrade package](#) for v26 from the [Pexip download page](#).
3. Before upgrading, ensure that all "always-on" Conferencing Nodes are powered on and are reachable (i.e. no Connectivity Loss errors), and are all running the same version from which you are upgrading. You do not need to power on any cloud bursting nodes.
4. From the Pexip Infinity Administrator interface, go to **Utilities > Upgrade**.
5. Select **Choose File** and browse to the location of the upgrade package.



6. Select **Continue**. There will be a short delay while the upgrade package is uploaded.
After the upgrade package has been uploaded, you are presented with a confirmation page showing details of the existing software version and the upgrade version.
7. To proceed, select **Start upgrade**.
You are taken to the **Upgrade Status** page, showing the current upgrade status of the Management Node and all Conferencing Nodes (for a definition of each status, see [Definition of upgrade statuses](#)). This page automatically refreshes every 5 seconds.
8. When the upgrade completes, all nodes will show a status of **No upgrade in progress** and have the new **Installed version**.
If a Conferencing Node fails to upgrade, for example if it remains on a **Waiting for calls to clear** status, it should be [rebooted](#). The upgrade process will then continue as expected.
 - ❗ If you are upgrading from v25.0 or v25.1, due to a known issue it is possible that the upgrade will complete on the Management Node but not automatically proceed to the Conferencing Nodes. To resolve this issue, simply upload the upgrade package again via **Utilities > Upgrade**.
9. If you have Pexip CVI for Microsoft Teams you must also upgrade your associated Teams Connector deployment in Azure to the same version as your Pexip Infinity deployment (including minor/"dot" releases).
 - ❗ When upgrading your Teams Connector to version 26:
 - You must be using Az module version 4.7.0 or later.
 - To check your installed version you can run:
`Get-InstalledModule -Name Az -AllVersions`
 - To install the latest Az version you can run:
`Install-Module -Name Az -MinimumVersion 4.7.0 -AllowClobber -Scope AllUsers`
 - You must use the latest version of the redeploy script as contained within the v26 documentation.
 - There are some important changes to the scripts in version 26:
 - The variables initialization script contains a new **\$tags** variable. This provides the ability to optionally specify a set of tags (name-value pairs) to apply to the Azure resources that are created for the Teams Connector.
 - The installation and redeploy/upgrade scripts have both been updated to use the new **\$tags** variable.
 - The redeploy/upgrade script now removes the dynamic Azure resource groups for you (those named **<prefix>-TeamsConn-<region>-RG** that contain the existing Teams Connector instances). This means you no longer have to perform this step manually via the Azure portal when upgrading or redeploying.
 - If you have deployed multiple Teams Connectors, you must follow the same redeploy process (with the appropriate variable initialization script) for each Teams Connector.

- As with all upgrades, you can continue to use the Pexip CVI app from your existing deployment.

Full instructions are available at https://docs.pexip.com/admin/teams_managing.htm#upgrading.

- i** No additional actions are required to upgrade the Management Node or Conferencing Nodes individually.
- i** We recommend that you take a fresh [backup](#) of your system after upgrading.

Error messages during upgrade

The following Error and Warning messages are expected during an upgrade and do not require any action:

```
2017-03-03T12:12:02.400+00:00 <manager_hostname> 2017-03-03 12:12:02,456 Level="ERROR"
Name="administrator.system.connectivity" Message="Unable to contact node." Src-Node="<node_ip_fqdn>"
Node="<node_ip_fqdn>"
```

```
2017-03-09T07:04:08.460+00:00 <manager_hostname> 2017-03-09 07:04:08,460 Level="WARNING"
Name="administrator.alarm" Message="Alarm raised" Node="<node_ip_fqdn>"
Alarm="connectivity_lost" Time="2017-03-09 07:04:08,455" Instance="Source=<node_ip_fqdn>,
Destination=<node_ip_fqdn>" Detail=""
```

Upgrading from versions 16-21 to version 26

If you are running a Pexip Infinity software version between v16 and v21 inclusive, you must first upgrade to version 22 and then upgrade again to version 26. To do this:

1. Before upgrading, ensure that all "always-on" Conferencing Nodes are powered on and are reachable (i.e. no Connectivity Loss errors), and are all running the same version from which you are upgrading. You do not need to power on any cloud bursting nodes (unless you are upgrading from version 21.0, in which case they must also be powered on at least 15 minutes prior to upgrading from v21.0).
2. Download the Pexip Infinity **v22** [upgrade file](#).
3. Follow the steps outlined in [Upgrading from version 22 or later to version 26](#), but when asked to **Choose File** browse to the location of the **v22** upgrade file.
4. Verify that the upgrade has completed successfully.
5. Download the Pexip Infinity **v26** upgrade file.
6. Follow the steps outlined in [Upgrading from version 22 or later to version 26](#), and when asked to **Choose File** browse to the location of the **v26** upgrade file.

Upgrading from versions 13-15 to version 26

If you are running a Pexip Infinity software version between v13 and v15 inclusive, you must first upgrade to version 17, then upgrade again to version 22, and then finally upgrade to version 26. To do this:

1. Download the Pexip Infinity **v17** [upgrade file](#).
2. Follow the steps outlined in [Upgrading from version 22 or later to version 26](#), but when asked to **Choose File** browse to the location of the **v17** upgrade file.
3. Verify that the upgrade has completed successfully and then follow the steps outlined above in [Upgrading from versions 16-21 to version 26](#).

Upgrading configuration-only deployments

The [automatic upgrade process](#) described above will update the Management Node and all Conferencing Nodes, including Conferencing Nodes that have been created using the [configuration only](#) deployment type.

However, if after the upgrade you wish to deploy new configuration-only Conferencing Nodes, you must download and use a version of the Conferencing Node VM template that matches the version of the Management Node that you have upgraded to. Creating a new Conferencing Node from a VM template containing a different version of the Pexip Infinity software than that which is running on the Management Node is not supported and will not work. For more information, see [Deploying a Conferencing Node using a generic VM template and configuration file](#).

Definition of upgrade statuses

During an upgrade, the **Upgrade Status** page will report the status of each Conferencing Node as follows:

Status	Description
No upgrade in progress	During a platform upgrade, this is the default state that occurs before a Conferencing Node is upgraded, or after the node has rebooted.
Upgrade pending	An upgrade of the platform is in progress and this Conferencing Node is in the queue to be upgraded.
Preparing to upgrade	The Conferencing Node is preparing to upgrade. During this time the node is put into maintenance mode , and other services are stopped.
Waiting for calls to clear	The Conferencing Node is in maintenance mode and is waiting for existing conferences to complete.
Timeout waiting for calls to clear	Not all conferences had cleared after 1 hour. This Conferencing Node has been removed from maintenance mode and the upgrade will be attempted again later.
Upgrade in progress	The new software is being unpackaged and installed on the Conferencing Node. During this time the Conferencing Node will not synchronize configuration with the Management Node.
Rebooting	The upgrade has completed and the Conferencing Node is rebooting.
Could not communicate with conferencing node	This error is reported if the Conferencing Node cannot be contacted.

Downgrading or recovering from a failed upgrade

To downgrade your system to the previous version, or to recover your system after a failed upgrade:

1. Restore the Management Node from the VM snapshot that you took using your hypervisor at the start of the upgrade process.
2. Restore the individual Conferencing Nodes by either:
 - redeploying each Conferencing Node from the Management Node - see [Deploying new Conferencing Nodes](#), or
 - restoring the hypervisor snapshot of each Conferencing Node that was taken at the start of the upgrade process.

Checking the installed version

Management Node

To view which software version is running on the Management Node, click on the **About** link at the top right corner of the Pexip Infinity Administrator interface.

Conferencing Nodes

To view which software version is running on individual Conferencing Nodes, go to **Status > Conferencing Nodes**. The software version number will be shown in the **Installed version** column.

Setting and changing usernames and passwords

There are a number of different components of the Pexip Infinity platform that require a username and password:

- Pexip Infinity Administrator interface
- Management Node operating system
- Conferencing Node operating system

Information on how to set and change these is given below.

Some external systems such as CUCM also use usernames and passwords to authenticate with Pexip Infinity. For more information on these, see [Integrating with external systems](#).

Pexip Infinity Administrator interface

A username and password are required for logging in to the Pexip Infinity Administrator interface. By default these credentials are verified against the local database and are the **web administration username** and **web administration password** that were set during initial installation (via the installation wizard). You can also configure the system to [authenticate users against an LDAP-accessible database](#).

Local database authentication

The username (typically `admin`) cannot be changed after being set, but the password can.

To change the password used to log in to the Pexip Infinity Administrator interface:

1. Log in to the Pexip Infinity Administrator interface.
2. At the top right of the screen, select **Change password**.
3. Enter the existing password, and then enter the new password twice.
4. Select **Change my password**.

You will see a message:

Password change successful

Your password was changed.

Note that the **Change password** option is not available if the system has its **Authentication source** configured as **LDAP database**.

LDAP-accessible database authentication

Username and passwords for the LDAP-accessible database cannot be managed via the Pexip Infinity Administrator interface.

Changing the Management Node operating system password

The Management Node virtual machine (VM) runs on a Linux operating system. The username is `admin` and cannot be changed.

The password for the Management Node operating system was set during initial installation of the Management Node (when running the installation wizard).

The password can be changed by logging in to the Management Node operating system over SSH (providing SSH access has not been disabled) and running the standard UNIX command `passwd`.

Conferencing Node operating system password

Conferencing Nodes run on a Linux operating system. The username for each Conferencing Node operating system is `admin` and cannot be changed.

The password was set during [Deploying new Conferencing Nodes](#) when configuring the **SSH password** field.

The password can be changed by logging in to the Conferencing Node operating system over SSH (providing SSH access has not been disabled) and running the standard UNIX command `passwd`.

Lost or forgotten passwords

If you forget the password for the Pexip Infinity Administrator interface, you can [re-run the installation wizard](#), being sure to change only the **Web administration password** setting.

i If you re-run the installation wizard to only reset the web administration password then you will not lose your configuration data and you will not need to delete and redeploy your Conferencing Nodes, providing you re-enter exactly the same hostname/addressing information as before. However, after completing the wizard you must then edit every system location (**Platform > Locations**) and reselect the DNS and NTP servers for that location.

If you forget the password used to log in to a Conferencing Node or Management Node operating system, contact your Pexip authorized support representative for assistance.

Re-running the installation wizard

You can rerun the installation wizard if you previously exited the wizard before completing all the steps, or you entered the wrong hostname/addressing information (for server-based deployments) during the initial configuration.

- i** Re-running the installation wizard will overwrite any existing Management Node configuration and lose all connections to existing Conferencing Nodes. Therefore, if you re-run the installation wizard after deploying any Conferencing Nodes, you must delete any existing Conferencing Nodes and then redeploy them from the updated Management Node.
- i** If you interrupt the installation wizard, or it does not complete properly for any reason, you must reboot the Management Node and then rerun the installation wizard.
- i** If you want to change the IP address of the Management Node you must first return your licenses from your current Management Node. If you do not return your licenses before re-running the installation wizard you will lose your licenses from your new Management Node (with the new IP address) and you will not be able to add them back to that new Management Node.
- i** Do not re-run the installation wizard on cloud-based deployments (Azure, AWS, GCP or Oracle) in order to change Management Node configuration data such as its IP address or hostname. To change such data you must terminate the existing instance and deploy a new Management Node instance. You should only re-run the installation wizard on cloud-based deployments if you need to reset the web administration password (and then you should not change any of the other configuration data).
- i** If you re-run the installation wizard to only reset the web administration password then you will not lose your configuration data and you will not need to delete and redeploy your Conferencing Nodes, providing you re-enter exactly the same hostname/addressing information as before. However, after completing the wizard you must then edit every system location (**Platform > Locations**) and reselect the DNS and NTP servers for that location.

To re-run the installation wizard:

1. Open a console window on the Management Node VM.
If your Management Node is deployed on Azure, AWS, GCP or Oracle, use an SSH client to access the Management Node, supplying your private key file if appropriate.
2. At the login prompt, log in as **admin**.
3. If requested, enter the operating system password created when the installation wizard was run. If you did not get as far as creating a new password, you are asked to create one now.
4. Type `installwizard`.
If your Management Node is deployed on Azure, AWS, GCP or Oracle, you are prompted again for the admin password.
The Pexip installation wizard begins.
(If you interrupt the installation wizard, or it does not complete properly for any reason, you must reboot the Management Node and then rerun the installation wizard.)

(Note that the steps described above can vary slightly depending on whether it is a cloud or server-based deployment, and whether you previously completed all of the installation wizard steps.)

Follow the prompts to set the following configuration for the Management Node.

If you press enter, the default value will be applied:

Setting	Default value	Multiple entries allowed?	Can be changed via Pexip Infinity Administrator interface?
IP address	192.168.0.100 *	No	No ‡
Network mask	255.255.255.0 *	No	No ‡
Gateway	192.168.0.1 *	No	No ‡
Hostname	<no default>	No	No ‡
Domain suffix	<no default>	No	No ‡
DNS servers	8.8.8.8	Yes, if separated by a space	Yes

Setting	Default value	Multiple entries allowed?	Can be changed via Pexip Infinity Administrator interface?
NTP servers †	Two of the following: <ul style="list-style-type: none"> 0.pexip.pool.ntp.org 1.pexip.pool.ntp.org 2.pexip.pool.ntp.org 3.pexip.pool.ntp.org 	Yes, if separated by a space	Yes
Web administration username	admin	No	No ‡
Web administration password	<no default>	No	Yes
Enable incident reporting (yes/no)	<no default>		Yes
Send deployment and usage statistics to Pexip (yes/no)	<no default>		Yes

* These defaults apply only if your network does not have DHCP enabled. If DHCP is enabled, the defaults are allocated by the DHCP server. The addresses entered here will be assigned as static IP addresses. When deploying in a cloud service, these values are replaced with the IP address and network settings for your instance.

† The NTP server must be accessible by the Management Node at the time the startup wizard is run. Installation will fail if the Management Node is unable to synchronize its time with an NTP server.


‡ After they have been configured, do not attempt to change these settings by any other means. To change these settings on server-based deployments, you must re-run the installation wizard. To change these settings on cloud-based deployments, you must terminate the existing instance and deploy a new Management Node instance.

Migrating Conferencing Nodes between host VMware servers

Using the tools available in VMware, you can move Conferencing Nodes from one host server to another.

You may want to do this if the existing host server is running low on capacity or is experiencing performance issues.

Pexip Infinity supports VMware's vMotion for the live migration of Conferencing Nodes.

 You must put the Conferencing Node into maintenance mode and wait until all conferences on that node have finished before migrating it to another host server.

Prerequisites

The source and target servers must be enabled for vMotion and use the same shared storage.

Manual migration via VMware's vMotion


1. Ensure that the Conferencing Node is in maintenance mode and that all conferences on that node have finished.
2. In the VMware inventory view, right-click on the Conferencing Node and select **Migrate**. The **Migrate Virtual Machine** window will appear.
3. Select **Change host** then select **Next**.
4. Select the destination host server or cluster and then select **Next**.
5. Select the destination resource pool and then select **Next**.
6. Select the priority and then select **Next**.
7. Check the details of the migration, and if they are all correct, select **Finish**.

Taking a Conferencing Node out of service

If you need to take a Conferencing Node out of service, you must first put it into maintenance mode to ensure that it is not hosting any conferences.

When a Conferencing Node goes into maintenance mode, it will not accept any new calls or registrations. Any existing calls on that Conferencing Node will not be affected (this applies to all call processing i.e. handling signaling, media proxying and transcoding). After all existing calls have terminated, the Conferencing Node will still be live and contactable but will not be handling any calls.

A Conferencing Node will go into maintenance mode if:

- The system administrator elects to put the Conferencing Node into maintenance mode via the Pexip Infinity Administrator interface (see [Manually placing a Conferencing Node into maintenance mode](#) below).
 -  This setting will persist after a reboot.
- The Pexip Infinity platform is being upgraded, during which time each Conferencing Node in turn is automatically put into maintenance mode and upgraded. For more information, see [Upgrading the Pexip Infinity platform](#).
- The Conferencing Node has been installed on a system that does not meet the CPU instruction set requirements. In such a case:
 - after initial deployment, the following message will appear in the admin log: `CPU instruction set is not supported; system will be placed in maintenance mode`
 - each time the Conferencing Node rejects a call, the following message will appear in the admin log: `Message="Participant failed to join conference." Reason="System in maintenance mode"`
 - any manual changes to the **Enable maintenance mode** setting will have no impact - the Conferencing Node will remain in maintenance mode regardless of this setting.

For more information on how to resolve this, see [Troubleshooting the Pexip Infinity platform](#).

- The Conferencing Node is running on a VM in Azure which has a scheduled maintenance event.

Note that when a Conferencing Node is in maintenance mode, it reports a media load of 100%. This is to indicate that there is no current capacity available.

Manually placing a Conferencing Node into maintenance mode

When putting multiple nodes into maintenance mode, to ensure maximum availability for registration data and other data that may have replicas stored in the distributed database on that node, and to ensure maximum service availability, we recommend that you:

- Activate maintenance mode on one node at a time.
- Wait a couple of minutes before putting the next node into maintenance mode.

This allows registrations and any relevant distributed data to gradually be migrated to other nodes.

To manually put a Conferencing Node into maintenance mode:

1. Go to **Platform > Conferencing Nodes**.
2. Select the Conferencing Node(s).
3. From the **Action** menu at the top left of the screen, select **Enable maintenance mode** and then select **Go**.
While maintenance mode is enabled, this Conferencing Node will not accept any new conference instances.
4. Wait until any existing conferences on that Conferencing Node have finished. To check, go to **Status > Live View**.

When all conferences on the Conferencing Node have finished you can take it out of service.

Long term maintenance

If you intend to take one or more nodes offline for a long period of time (days or weeks) then you should consider temporarily deleting those nodes altogether — and redeploying them afresh when they are next needed. This will avoid distracting connectivity alarms being raised for those nodes that might make it harder for the administrator to spot other "unexpected" alarms in the meantime.

Detecting maintenance mode

Load balancers can use the `https://<node_address>/api/client/v2/status` REST API command to check whether a Conferencing Node is in maintenance mode.

Rebooting and shutting down a Conferencing Node

Occasionally, it may be necessary to reboot or shut down a Conferencing Node.

Rebooting a Conferencing Node

In many cases, Conferencing Nodes will reboot automatically as a result of certain configuration updates:

- Changing some specific **Global Settings** such as the media and signaling port ranges, or when you have enabled or disabled call protocols.
- Changing the system location of a Conferencing Node.
- Changing a Conferencing Node's role from a transcoding node to a proxying node or vice versa.

However, sometimes a manual reboot may be required, for example if a Conferencing Node fails to upgrade and it remains on a **Waiting for calls to clear status**.

There are two main ways to reboot a Conferencing Node:

- Log in to the hypervisor or your cloud service that is managing the host server and reboot the virtual machine using the relevant processes according to the management system.
- Connect to the Conferencing Node via SSH, log in as admin, and issue the command `sudo reboot`.

Shutting down a Conferencing Node

Sometimes you may need to shut down a Conferencing Node for maintenance, or if you are deleting it permanently. There are two main ways to do this:

- Log in to the hypervisor that is managing the host server and power off the virtual machine using the relevant processes according to the management system.
- Connect to the Conferencing Node via SSH, log in as admin, and issue the command `sudo poweroff`.


If your nodes are running in a cloud service, you should power off via SSH / serial console for a clean shutdown, and then use the cloud service's management portal to shutdown/stop the virtual machine to ensure that its associated resources are released.

Moving, restoring or changing the IP address of the Management Node

These procedures explain how to change the IP address of your Management Node, or physically move your Management Node to a new host server, without losing your Pexip Infinity platform configuration or invalidating your Pexip Infinity licenses.

Before performing any of these procedures we recommend that you follow standard IT practices, as per your company policy. For example, ensure that you take a snapshot of the Management Node prior to making any changes, and if possible take offline backups of the VMs.

In all cases you will have a loss of service while you perform the maintenance procedure.


-  Note that these instructions only apply to on-premises deployments. For cloud-based deployments, if you need to change the IP address of the Management Node you must decommission your entire deployment, terminating all of the existing instances, and then redeploy your system.

Changing the IP address of the Management Node (same physical host)

To change the IP address of your Management Node, where you want to keep the same physical host for the Management Node:

1. We recommend that you backup your current configuration:
 - a. Go to **Utilities > Backup/Restore**.
 - b. In the **Create backup** section, enter a **Passphrase** and then enter it again in the **Re-enter passphrase** field.
The text entered here is used to encrypt the backup file. You must remember this text as it will be required if you need to subsequently restore the data from the file.
 - c. Select **Create backup**.
After a few seconds you see a message: "Successfully created the backup file: <file_name>" where <file_name> takes the format:
`pexip_backup_<hostname>_<version>_<build>_<date>_<time>.tar.pexbak`

- d. Download the file from the host VM:
 - i. From the **Existing backup files** section at the bottom of the page, select **Download backup** for the file you have just created.
 - ii. Follow your browser's prompts to save or download the file to your local file system.
 - iii. If required, you can delete unwanted backup files from your host VM by selecting **Delete backup**.

The system keeps on the host VM only the 5 most recent manually-taken backups and the 5 most recent automatic backups. Older backup files are deleted.
2. Return your Pexip Infinity license.
 -  You must return your license before decommissioning your current host installation.
 - a. Go to **Platform > Licenses**.
 - b. Select the license you want to deactivate.
 - c. Make a note of the **License entitlement key**.
 - d. Select **Return license**.
3. Decommission your Conferencing Nodes i.e. delete the Conferencing Nodes and remove the associated VMs. Follow this procedure for each Conferencing Node:
 - a. Put the Conferencing Node into **maintenance mode** and wait until all calls on it have terminated.
 - b. From the Pexip Infinity Administrator interface, go to **Platform > Conferencing Nodes**.
 - c. Select the Conferencing Node to be deleted, and from the **Action** drop-down menu select **Delete selected Conferencing Nodes**.
 - d. Select **Go** and on the following page confirm that you want to delete the selected Conferencing Node by selecting **Yes, I'm sure**.
 - e. Shut down and remove the Conferencing Node VM.
 - i. Log in to the VM manager, shut down the deleted Conferencing Node and then power it off.
 - ii. Right-click on the Conferencing Node and select **Delete from Disk** (VMware) or **Delete** (Hyper-V / KVM / Xen).
4. Re-run the installation wizard and supply the new IP address of the Management Node:
 - a. Open a console window on the Management Node VM.
 - b. At the login prompt, log in as **admin**.
 - c. Enter the operating system password.
 - d. Type `installwizard`.

The Pexip installation wizard begins.

At the IP address prompt, enter your new Management Node IP address.

(If you interrupt the installation wizard, or it does not complete properly for any reason, you must reboot the Management Node and then rerun the installation wizard.)
5. Reapply your previous license:
 - a. Go to **Platform > Licenses**.
 - b. Select **Add License**.
 - c. In the **License entitlement key** field, enter the entitlement key from the 'old' Management Node that you noted previously.
 - d. Select **Save**.

The system will attempt to activate the license on the 'new' Management Node:

 - If the license is activated successfully, you are returned to the **Licensing** page and the new license is shown under the **Licensing** section.
 - If the activation attempt is unsuccessful (for example, if the Management Node was unable to establish a connection to the Pexip licensing server), or you selected **Manually activate**, the license is saved as a **Stored license request**. You must then [activate it manually](#).
6. Re-provision your Conferencing Nodes.

See [Deploying new Conferencing Nodes](#) for links to the appropriate instructions for your hypervisor.

Moving/restoring a Management Node to a new physical host (same IP address)

Note that if your Management Node is installed on VMware and you do not need to perform a fundamental redeployment of the Management Node, i.e. you only want to move it across host servers, then you should use vMotion to perform the move.


To move or restore a Management Node to a new physical host, where you are keeping the same IP address for the Management Node on the new host server:

1. Backup your current configuration:
 - a. Go to **Utilities > Backup/Restore**.
 - b. In the **Create backup** section, enter a **Passphrase** and then enter it again in the **Re-enter passphrase** field.

The text entered here is used to encrypt the backup file. You must remember this text as it will be required if you need to subsequently restore the data from the file.
 - c. Select **Create backup**.

After a few seconds you see a message: "Successfully created the backup file: <file_name>" where <file_name> takes the format:

```
pexip_backup_<hostname>_<version>_<build>_<date>_<time>.tar.pexbak
```
 - d. Download the file from the host VM:
 - i. From the **Existing backup files** section at the bottom of the page, select **Download backup** for the file you have just created.
 - ii. Follow your browser's prompts to save or download the file to your local file system.
 - iii. If required, you can delete unwanted backup files from your host VM by selecting **Delete backup**.

The system keeps on the host VM only the 5 most recent manually-taken backups and the 5 most recent automatic backups. Older backup files are deleted.
2. Return your Pexip Infinity license.
 -  You must return your license before decommissioning your current host installation.
 - a. Go to **Platform > Licenses**.
 - b. Select the license you want to deactivate.
 - c. Make a note of the **License entitlement key**.
 - d. Select **Return license**.
3. Install the Management Node on the new host server.

Note that all of the configuration data that you enter via the installation wizard, including the IP address, will be subsequently replaced when you restore your backup.

See [Installation overview](#) for links to the appropriate instructions for your hypervisor.
4. Restore the previous configuration to the new Management Node from your backup.
 - a. Go to **Utilities > Backup/Restore**.
 - b. In the **Restore backup** section, enter the **Passphrase**.

The text entered here must be identical to the text that was used to create the backup file.
 - c. Select **Choose File** and then choose the backup file that you want to restore.

The file must be chosen from your local file system (you cannot select a file from the list of **Existing backup files**).
 - d. Select **Restore backup**.

You are taken to the **Restore Backup** confirmation page.

If instead, you see "Failed to restore the system from the backup file" with a "Decryption Error: decryption failed" message, the most likely reason for this is that you have entered an incorrect passphrase.
 - e. The confirmation page shows the date that the backup was taken, and the Management Node IP address that will be restored to this system.

Select **Restore backup** to confirm the restoration.

If the restore is successful, after a few seconds you will see a "Successfully restored the backup file" message and the Management Node will reboot.

You need to wait for the node to return from rebooting before you can access it again.

5. Reapply your previous license:
 - a. Go to **Platform > Licenses**.
 - b. Select **Add License**.
 - c. In the **License entitlement key** field, enter the entitlement key from the 'old' Management Node that you noted previously.
 - d. Select **Save**.

The system will attempt to activate the license on the 'new' Management Node:

- If the license is activated successfully, you are returned to the **Licensing** page and the new license is shown under the **Licensing** section.
- If the activation attempt is unsuccessful (for example, if the Management Node was unable to establish a connection to the Pexip licensing server), or you selected **Manually activate**, the license is saved as a **Stored license request**. You must then [activate it manually](#).

Moving/restoring a Management Node to a new physical host (different IP address)


To manually move or restore a Management Node to a new physical host server, where you also want to use a different IP address for the Management Node:

1. Decommission your Conferencing Nodes i.e. delete the Conferencing Nodes and remove the associated VMs. Follow this procedure for each Conferencing Node:
 - a. Put the Conferencing Node into [maintenance mode](#) and wait until all calls on it have terminated.
 - b. From the Pexip Infinity Administrator interface, go to **Platform > Conferencing Nodes**.
 - c. Select the Conferencing Node to be deleted, and from the **Action** drop-down menu select **Delete selected Conferencing Nodes**.
 - d. Select **Go** and on the following page confirm that you want to delete the selected Conferencing Node by selecting **Yes, I'm sure**.
 - e. Shut down and remove the Conferencing Node VM.
 - i. Log in to the VM manager, shut down the deleted Conferencing Node and then power it off.
 - ii. Right-click on the Conferencing Node and select **Delete from Disk** (VMware) or **Delete** (Hyper-V / KVM / Xen).
2. Backup your current configuration:
 - a. Go to **Utilities > Backup/Restore**.
 - b. In the **Create backup** section, enter a **Passphrase** and then enter it again in the **Re-enter passphrase** field.

The text entered here is used to encrypt the backup file. You must remember this text as it will be required if you need to subsequently restore the data from the file.
 - c. Select **Create backup**.

After a few seconds you see a message: "Successfully created the backup file: <file_name>" where <file_name> takes the format:

```
pexip_backup_<hostname>_<version>_<build>_<date>_<time>.tar.pexbak
```
 - d. Download the file from the host VM:
 - i. From the **Existing backup files** section at the bottom of the page, select **Download backup** for the file you have just created.
 - ii. Follow your browser's prompts to save or download the file to your local file system.
 - iii. If required, you can delete unwanted backup files from your host VM by selecting **Delete backup**.

The system keeps on the host VM only the 5 most recent manually-taken backups and the 5 most recent automatic backups. Older backup files are deleted.
3. Return your Pexip Infinity license.
 -  You must return your license before decommissioning your current host installation.
 - a. Go to **Platform > Licenses**.
 - b. Select the license you want to deactivate.
 - c. Make a note of the **License entitlement key**.
 - d. Select **Return license**.

4. Install the Management Node on the new host server.

All the configuration data you enter via the installation wizard, including the IP address, will be subsequently replaced when you restore your backup. Thus, at this stage you can use the same IP address as you used for your previous Management Node installation.

See [Installation overview](#) for links to the appropriate instructions for your hypervisor.

5. Restore the previous configuration to the new Management Node from your backup (this also restores the original Management Node IP address — you set the new address in the next step):

- a. Go to **Utilities > Backup/Restore**.

- b. In the **Restore backup** section, enter the **Passphrase**.

The text entered here must be identical to the text that was used to create the backup file.

- c. Select **Choose File** and then choose the backup file that you want to restore.

The file must be chosen from your local file system (you cannot select a file from the list of **Existing backup files**).

- d. Select **Restore backup**.

You are taken to the **Restore Backup** confirmation page.

If instead, you see "Failed to restore the system from the backup file" with a "Decryption Error: decryption failed" message, the most likely reason for this is that you have entered an incorrect passphrase.

- e. The confirmation page shows the date that the backup was taken, and the Management Node IP address that will be restored to this system.

Select **Restore backup** to confirm the restoration.

If the restore is successful, after a few seconds you will see a "Successfully restored the backup file" message and the Management Node will reboot.

You need to wait for the node to return from rebooting before you can access it again.

6. Re-run the installation wizard and supply the new IP address of the Management Node:

- a. Open a console window on the Management Node VM.

- b. At the login prompt, log in as **admin**.

- c. Enter the operating system password.

- d. Type `installwizard`.

The Pexip installation wizard begins.

At the IP address prompt, enter your new Management Node IP address.

(If you interrupt the installation wizard, or it does not complete properly for any reason, you must reboot the Management Node and then rerun the installation wizard.)

7. Reapply your previous license:

- a. Go to **Platform > Licenses**.

- b. Select **Add License**.

- c. In the **License entitlement key** field, enter the entitlement key from the 'old' Management Node that you noted previously.

- d. Select **Save**.

The system will attempt to activate the license on the 'new' Management Node:

- If the license is activated successfully, you are returned to the **Licensing** page and the new license is shown under the **Licensing** section.
- If the activation attempt is unsuccessful (for example, if the Management Node was unable to establish a connection to the Pexip licensing server), or you selected **Manually activate**, the license is saved as a **Stored license request**. You must then [activate it manually](#).

8. Re-provision your Conferencing Nodes.

See [Deploying new Conferencing Nodes](#) for links to the appropriate instructions for your hypervisor.

Bulk import/export of service configuration data

If you want to configure your Pexip Infinity platform with a large number of Virtual Meeting Rooms, Virtual Auditoriums, Virtual Receptions, Automatically Dialed Participants (APDs) or device aliases, you can import the configuration for each of these items from a CSV file.

You can also export all of your existing service configuration data to a CSV file. You may want to do this for backup purposes or to transfer configuration between, for example, a test system and a production system.

You can also use a CSV file to import multiple endpoints for One-Touch Join.

- i** You can import Virtual Meeting Rooms and device aliases from directory information contained in a Windows Active Directory LDAP server, or any other LDAP-accessible database. For more information, see [Provisioning VMRs, devices and users from Active Directory via LDAP](#).

Preparing the CSV file for import

When formatting your import file:

- A header row in the CSV file is optional. If included, it must use the same field names as shown in the following sections, but you may change the order of the fields. If a header row is not used, fields must be in the same order as shown for each import file type.
- All non-blank fields must contain valid data, for example the pin field must only contain digits and the `dtmf_sequence` field must only contain digits or commas.
- If non-ASCII characters are used, the file must be encoded as UTF-8 text.
- Fields with values of `true` or `false` are not case-sensitive. All other fields are case-sensitive.
- Values may optionally be enclosed in double quotation marks; any strings containing commas must be enclosed in double quotation marks e.g. "description for x, y and z".

Note that you can perform an export of existing data to produce an example file in the correct format.

The sections below contain the specific field requirements for your [service](#), [device alias](#) or [ADP](#) import file.

Services (Virtual Meeting Rooms, Virtual Auditoriums and Virtual Receptions)

This section explains how to format a CSV import file of Virtual Meeting Rooms, Virtual Auditoriums or Virtual Receptions, and how to perform an [import](#) or [export](#).

This allows you to import all configuration apart from the Automatically Dialed Participants (ADPs) to be associated with each service. These can be imported separately; see [Automatically dialed participants \(ADPs\)](#) below.

You must prepare separate CSV files for Virtual Meeting Rooms, Virtual Auditoriums and Virtual Receptions, using the fields as shown below.

Virtual Meeting Room file format

```
name,description,pin,allow_guests,guest_pin,alias_alias,alias_description,tag,max_callrate_in,max_callrate_out,call_type,host_view,enable_overlay_text,ivr_theme_name,participant_limit,primary_owner_email_address,guests_can_present,enable_chat,crypto_mode,max_pixels_per_second,enable_active_speaker_indication
```


Virtual Auditorium file format

```
name,description,pin,allow_guests,guest_pin,alias_alias,alias_description,tag,max_callrate_in,max_callrate_out,call_type,host_view,guest_view,force_presenter_into_main,enable_overlay_text,ivr_theme_name,participant_limit,mute_all_guests,guests_can_present,enable_chat,crypto_mode,max_pixels_per_second
```

Virtual Reception file format

```
name,description,alias_alias,alias_description,tag,max_callrate_in,max_callrate_out,call_type,ivr_theme_name,mssip_proxy_name,teams_proxy_name,match_string,replace_string,system_location_name,post_match_string,post_replace_string,two_stage_dial_type,gms_access_token_name,crypto_mode,max_pixels_per_second
```

where:

Field name	Content
name	<p>The name used to refer to this Virtual Meeting Room, Virtual Auditorium or Virtual Reception.</p> <p>This field cannot be left blank.</p> <p> If you can access this service via a Virtual Reception then the service Name entered here is shown to conference participants as they are transferred into the service (it is overlaid onto the <code>virtual_reception_connecting</code> splash screen of the theme associated with the Virtual Reception that is transferring the call).</p>
description	An optional description of the Virtual Meeting Room, Virtual Auditorium or Virtual Reception.
pin	<p>(Virtual Meeting Room and Virtual Auditorium only)</p> <p>This optional field allows you to set a secure access code that must be entered by participants before they can join the conference.</p> <p>If a Guest PIN has also been set, then the PIN will apply to the conference Host(s) only.</p> <p>For more information, see About PINs, Hosts and Guests.</p>
allow_guests*	<p>(Virtual Meeting Room and Virtual Auditorium only)</p> <p>Determines whether the conference will allow participants with Guest privileges. For more information, see About PINs, Hosts and Guests.</p> <ul style="list-style-type: none"> • true: the conference has two types of participants: Hosts and Guests. The pin to be used by Hosts must be specified. A <code>guest_pin</code> can optionally be specified; if a <code>guest_pin</code> is not specified, Guests can join without a PIN. • false: all participants have Host privileges <p>Default: false</p>
guest_pin	<p>(Virtual Meeting Room and Virtual Auditorium only)</p> <p>This optional field allows you to set a secure access code that must be entered by Guests before they can join the conference.</p> <p>For more information, see About PINs, Hosts and Guests.</p>
alias_alias	<p>The alias that, when received by Pexip Infinity, will cause it to route the call to this service (Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service).</p> <p>The alias entered here must match the alias as it is received by Pexip Infinity. Wildcards and regular expressions are not supported.</p> <p>In most cases, the alias received by Pexip Infinity will be the same as the alias that the conference participant dialed from their endpoint, but there are some exceptions, described in About aliases and access numbers.</p> <p>You may also want to define multiple aliases for the same service to ensure that it can be accessed by devices and protocols that enforce specific alias formats — for more information, see Using multiple aliases to access the same service.</p>
alias_description	An optional description of the alias. This is useful if you have more than one alias for a service. Note that this description may be displayed to end users on registered Infinity Connect clients who are performing a directory search.
tag	This optional field lets you assign a unique identifier to this service, which you can then use to track use of the service .
max_callrate_in	<p>The maximum media bandwidth in kbps that Pexip Infinity will receive from each individual participant dialed in to the service. Range 128 to 4096 kbps.</p> <p>For more information see Managing and restricting call bandwidth.</p>

Field name	Content
max_callrate_out	<p>The maximum media bandwidth in kbps that Pexip Infinity will send to each individual participant dialed in to the service. Range 128 to 4096 kbps.</p> <p>For more information see Managing and restricting call bandwidth.</p>
call_type *	<p>Allows you to limit the media content of the conference. For more information, see Controlling media capability. Valid values are:</p> <ul style="list-style-type: none"> "video": main video plus presentation "video-only": main video only "audio": audio-only <p>Default: "video"</p>
host_view *	<p>(Virtual Meeting Room and Virtual Auditorium only)</p> <p>The maximum number of other participants that each host participant will see, and the layout used to show them. For more information, see Selecting the layout seen by participants. Valid values are:</p> <ul style="list-style-type: none"> "one_main_zero_pips": full-screen main speaker only "one_main_seven_pips": large main speaker and up to 7 other participants "one_main_twentyone_pips": main speaker and up to 21 other participants "two_mains_twentyone_pips": two main speakers and up to 21 other participants "four_mains_zero_pips": up to four main speakers, with no thumbnails "five_mains_seven_pips": Adaptive Composition layout <p>Default: "one_main_seven_pips"</p>
guest_view *	<p>(Virtual Auditorium only)</p> <p>The maximum number of Host participants that each Guest participant will see, and the layout used to show them. (Guests will only see Hosts; they can hear but not see any of the other Guests.) For more information, see Selecting the layout seen by participants. Valid values are:</p> <ul style="list-style-type: none"> "one_main_zero_pips": full-screen main speaker only "one_main_seven_pips": large main speaker and up to 7 other participants "one_main_twentyone_pips": main speaker and up to 21 other participants "two_mains_twentyone_pips": two main speakers and up to 21 other participants "four_mains_zero_pips": up to four main speakers, with no thumbnails "five_mains_seven_pips": Adaptive Composition layout (you must also set Host view to Adaptive Composition) <p>Default: "one_main_seven_pips"</p>
force_presenter_into_main *	<p>(Virtual Auditorium only)</p> <p>When a presentation is being shown, this option controls whether the main speaker position shows the presenter or the current speaker. For more information, see Selecting the layout seen by participants.</p> <ul style="list-style-type: none"> true: the Host sending the presentation stream will always hold the main video position false: the main video position is voice-switched <p>Default: false</p>

Field name	Content
enable_ active_ speaker_ indication *	<p>(Virtual Meeting Room only)</p> <p>When active speaker display is enabled, the display name or alias of the current speaker is shown across the bottom of their video image. This option is only available in Adaptive Composition layouts.</p> <ul style="list-style-type: none"> true: active speaker is indicated false: active speaker is not indicated <p>Default: false</p>
enable_ overlay_text *	<p>(Virtual Meeting Room and Virtual Auditorium only)</p> <p>If participant name overlays are enabled, the display names or aliases of all participants are shown in a text overlay along the bottom of their video image.</p> <ul style="list-style-type: none"> true: participant names are shown false: participant names are not shown <p>Default: false</p>
ivr_theme_ name	<p>The name of the theme to use with the service. If no theme is specified, the default Pexip theme is used.</p> <p>For more information, see Customizing conference images and voice prompts using themes.</p>
participant_ limit	<p>This optional field allows you to limit the number of participants allowed to join this VMR. For more information see Limiting the number of participants.</p>
primary_ owner_email_ address	<p>(Virtual Meeting Room only)</p> <p>The email address of the owner of the VMR.</p>
mute_all_ guests *	<p>(Virtual Auditorium only)</p> <p>Determines whether Guest participants are muted when they first join the conference.</p> <ul style="list-style-type: none"> true: mute Guests when they first join the conference false: do not mute Guests when they first join the conference <p>Default: false</p>
guests_can_ present *	<p>Controls whether the Guests in the conference are allowed to present content.</p> <ul style="list-style-type: none"> true: Guests and Hosts can present into the conference false: only Hosts can present <p>Default: true</p>
enable_chat *	<p>(Virtual Meeting Room and Virtual Auditorium only)</p> <p>Determines whether chat message support is enabled in the conference. Valid values are:</p> <ul style="list-style-type: none"> "default": as per the global configuration setting "yes": chat is enabled "no": chat is disabled <p>Default: "default"</p>

Field name	Content
crypto_mode *	<p>Controls the media encryption requirements for participants connecting to this service.</p> <ul style="list-style-type: none"> <null>: use the global media encryption setting. "besteffort": each participant will use media encryption if their device supports it, otherwise the connection will be unencrypted. "on": all participants (including RTMP participants) must use media encryption. "off": all H.323, SIP and MS-SIP participants must use unencrypted media. (RTMP participants will use encryption if their device supports it, otherwise the connection will be unencrypted.) <p>Default: <null> (use global setting)</p>
max_pixels_per_second *	<p>Controls the maximum call quality for participants connecting to this service:</p> <ul style="list-style-type: none"> <null>: use the global maximum call quality setting. "sd": each participant is limited to SD quality. "hd": each participant is limited to HD (720p) quality. "fullhd": allows any endpoint capable of Full HD to send and receive its main video at 1080p. <p>Default: <null> (use global setting)</p>
mssip_proxy_name	<p>(Virtual Reception only)</p> <p>The name of the Skype for Business / Lync server to use to resolve the SfB/Lync Conference ID entered by the user.</p>
teams_proxy_name	<p>(Virtual Reception only)</p> <p>The name of the Teams Connector to use to resolve Microsoft Teams meeting codes. If you do not specify anything, the Teams Connector associated with the outgoing location is used.</p>
match_string	<p>(Virtual Reception only)</p> <p>An optional regular expression used to match against the alias entered by the caller into the Virtual Reception. If the entered alias does not match the expression, the Virtual Reception will not route the call.</p> <p>If this field is left blank, any entered alias is permitted.</p>
replace_string	<p>(Virtual Reception only)</p> <p>An optional regular expression used to transform the alias entered by the caller into the Virtual Reception. (Only applies if a regex match string is also configured and the entered alias matches that regex.)</p> <p>Leave this field blank if you do not want to change the alias entered by the caller.</p>
system_location_name	<p>(Virtual Reception only)</p> <p>This is an optional field used in conjunction with the <code>two_stage_dial_type</code> setting, when a type other than <i>regular</i> is selected. If specified, a Conferencing Node in this system location will perform the SfB/Lync Conference ID lookup on the SfB/Lync server, or the Microsoft Teams or Google Meet code lookup, as appropriate. We recommend that a location is specified here, otherwise the transcoding node hosting the Virtual Reception will perform the lookup (which may lead to routability issues).</p>
post_match_string	<p>(Virtual Reception only)</p>
post_replace_string	<p>(Virtual Reception only)</p>

Field name	Content
two_stage_ dial_type *	(Virtual Reception only) The type of Virtual Reception: <ul style="list-style-type: none"> "regular": the default type of Virtual Reception, used for routing calls to VMRs, or to other devices and call control systems via the Infinity Gateway. "mssip": a special type of Virtual Reception, used when you want to provide an IVR gateway to scheduled and ad hoc Skype for Business / Lync meetings. "gms": a special type of Virtual Reception, used when you want to provide an IVR gateway to scheduled and ad hoc Google Meet meetings. "teams": a special type of Virtual Reception, used when you want to provide an IVR gateway to scheduled and ad hoc Microsoft Teams meetings. Default: "regular"
gms_access_ token_name	(Virtual Reception only)
* If this field is left blank, the default value is used.	

Multiple aliases

For services with more than one alias, you must add an additional record for each additional alias that repeats all fields except the **alias_alias** and **alias_description**:

```
name1,description1,pin1,allow_guests,guest_pin1,alias_alias1,alias_description1,...
name1,description1,pin1,allow_guests,guest_pin1,alias_alias2,alias_description2,...
```

Duplicates

If any records in the CSV file have the same **name** field, and any of the other fields apart from **alias_alias** and **alias_description** are different, only one service with that name will be created. This service will use the last record that was imported.

If any records in the CSV file have the same **name** as an existing service, the existing configuration will be overwritten by the imported service's configuration.

Examples

To import a Virtual Meeting Room called **alice** with a single alias of **meet.alice**, and a second Virtual Meeting Room called **bob** with aliases **meet.bob** and **meet.bobby**, you would create the following CSV file:

```
alice,,,,,meet.alice
bob,,,,,meet.bob
bob,,,,,meet.bobby
```

To import Virtual Meeting Rooms for Alice, Bob and Charlie that each have different [Host and Guest PINs](#) you would create the following CSV file:

```
alice,,1234,True,6789,meet.alice
bob,,4567,True,9876,meet.bob
charlie,,5432145,True,5556789,meet.charlie
```

Importing the CSV file

Virtual Meeting Rooms, Virtual Auditoriums and Virtual Receptions require separate CSV files.

To import the data in the CSV file to Pexip Infinity:

- On the Pexip Infinity Administrator interface, go to either:
 - Services > Virtual Meeting Rooms** or
 - Services > Virtual Auditoriums** or

- **Services > Virtual Receptions.**
- 2. Select **Import**.
- 3. Choose the CSV file to import and select **Save**.

Exporting existing service configuration

You can export all of your existing Virtual Meeting Room, Virtual Auditorium or Virtual Reception configuration data to a CSV file. This produces a CSV file in the same format as that used for importing configuration data (as described above). The file includes a header row.

 This feature exports all configuration except the [Automatically Dialed Participants](#) associated with each service.

To export the data:

1. On the Pexip Infinity Administrator interface, go to either:
 - **Services > Virtual Meeting Rooms** or
 - **Services > Virtual Auditoriums** or
 - **Services > Virtual Receptions.**
2. Select **Export**. This takes you to the **Export Configuration** page.
3. Select **Download**.
4. Follow your browser's prompts to save or open the file.

Device aliases

This section explains how to format a CSV import file of device aliases, and how to perform an [import](#) or [export](#).

You must prepare the CSV file using the fields as shown below:

```
alias,tag,description,username,password,primary_owner_email_address
```

where:

Field name	Content
alias	The alias URI that a device/client can register to Pexip Infinity. The alias must be an exact match; regular expressions are not supported. It cannot be blank.
tag	This optional field lets you assign a unique identifier to this service, which you can then use to track use of the service .
description	An optional description of the device.
username	The username with which to authenticate the device. The username and password credentials are optional and the device is not challenged if no credentials are entered. These credentials do not apply when using Infinity Connect registration via SSO services.
password	The password with which to authenticate the device, in association with the username. When passwords are exported to a CSV file, they are encrypted. You can however re-import the encrypted form of the password and it will be saved as the original.
primary_owner_email_address	The email address of the owner of the device alias.

Duplicates

If any records in the CSV file have the same **alias** field, only one device alias with that name will be created. This device alias will use the last record that was imported.

If any records in the CSV file have the same **alias** as an existing device alias, the existing configuration will be overwritten by the imported device alias's configuration.

Examples

```
alias,tag,description,username,password,sync_tag,primary_owner_email_address
```

To create aliases that will allow Alice and Bob to register their devices to Pexip Infinity with a username and password, you would create the following CSV file:

```
alice@example.com,,alice,1234
bob@example.com,,bob,5678
```

Importing the CSV file

To import the data in the CSV file to Pexip Infinity:

1. On the Pexip Infinity Administrator interface, go to **Users & Devices > Device Aliases**.
2. From the bottom of the page, select **Import**.
3. Choose the CSV file to import and select **Save**.

Exporting existing device alias configuration

You can export all of your existing device alias configuration data to a CSV file. This produces a CSV file in the same format as that used for importing configuration data. The file includes a header row.

To export the data:

1. On the Pexip Infinity Administrator interface, go to **Users & Devices > Device Aliases**.
2. From the bottom of the page, select **Export**. This takes you to the **Export Device Alias Configuration** page.
3. Select **Download**.
4. Follow your browser's prompts to save or open the file.


Automatically dialed participants (ADPs)


This section explains how to format a CSV import file of Automatically dialed participants (ADPs), and how to perform an [import](#) or [export](#).

You must prepare the CSV file using the fields as shown below:

```
alias,remote_display_name,description,protocol,dtmf_sequence,role,streaming,keep_conference_alive,call_type,routing,presentation_url,system_location_name,conference_name
```

where:

Field name	Content
alias	The alias of the participant to dial when a conference starts. It cannot be blank.
remote_display_name	An optional friendly name for this participant. This may be used instead of the participant's alias in participant lists and as a text overlay in some layout configurations.
description	An optional description of the Automatically Dialed Participant.
protocol *	<p>The protocol to use to place the outgoing call:</p> <ul style="list-style-type: none">• "sip"• "h323"• "rtmp"• "mssip" (for calls to Microsoft Skype for Business / Lync)• "auto" (to use Call Routing Rules) <p>Default: sip</p> <p> The protocol is ignored if the routing field is set to routing_rule.</p>

Field name	Content
dtmf_sequence	<p>An optional DTMF sequence to transmit after the call to the dialed participant starts.</p> <p> If one or more commas are used in the DTMF sequence (as a 2-second pause), the entire string within the field must be contained in double quotes.</p>
role *	<p>The level of privileges the participant has in the conference:</p> <ul style="list-style-type: none"> "chair": the participant has Host privileges "guest": the participant has Guest privileges <p>Default: guest</p>
streaming *	<p>Identifies the dialed participant as a streaming or recording device.</p> <ul style="list-style-type: none"> true: the participant is a streaming or recording device false: the participant is not a streaming or recording device <p>Default: false</p>
keep_conference_alive *	<p>Determines whether the conference continues when all other non-ADP participants have disconnected:</p> <ul style="list-style-type: none"> "keep_conference_alive": the conference continues to run until this participant disconnects (applies to Hosts only). "keep_conference_alive_if_multiple": the conference continues to run as long as there are two or more "keep_conference_alive_if_multiple" participants and at least one of them is a Host. "keep_conference_alive_never": the conference terminates automatically if this is the only remaining participant. <p>Default: keep_conference_alive_if_multiple</p>
call_type *	<p>Allows you to limit the media content of the call. The participant being called will not be able to escalate beyond the selected capability. For more information, see Controlling media capability. Valid values are:</p> <ul style="list-style-type: none"> "video": main video plus presentation "video-only": main video only "audio": audio-only <p>Default: "video"</p>
routing *	<p>Specifies how to route the call:</p> <ul style="list-style-type: none"> "manual": uses the requested protocol and the defaults for the specified system_location_name. "routing_rule": routes the call according to the configured Call Routing Rules. This means that the dialed alias must match an outgoing Call Routing Rule for the call to be placed (using the protocols, outgoing location and call control systems etc. as configured for that rule). <p>Default: "manual"</p>
presentation_url	<p>This additional parameter can be specified for RTMP calls to send the presentation stream to a separate RTMP destination.</p>
system_location_name	<p>The location of the Conferencing Node from which to place the call.</p>
conference_name	<p>The name of the Virtual Meeting Room or Virtual Auditorium from which this participant will be dialed automatically whenever a conference using that service starts.</p>
* If this field is left blank, the default value is used.	

Duplicates

You can upload any number of records with the same **alias** field, as long as the **conference_name** field is different for each.

If any records in the CSV file have the same **alias** and **conference_name** fields, only one ADP for that conference name will be created. This ADP will use the last record that was imported.

If any records in the CSV file have the same `alias` and `conference_name` as an existing ADP, the existing configuration will be overwritten by the imported ADP's configuration.

Examples

To import an ADP for Alice's endpoint (`alice@example.com`) that will be dialed in as a Host when her VMR (`meet Alice`) is called, and as a Guest when the sales team's Virtual Auditorium (`meet Sales`) is called, you would create the following CSV file:

```
alice@example.com,Alice Jones,,,,chair,,,,,meet Alice
alice@example.com,Alice Jones,,,,guest,,,,,meet Sales
```

Importing the CSV file

To import the data in the CSV file to Pexip Infinity:

1. On the Pexip Infinity Administrator interface, go to **Services > Automatically Dialed Participants**.
2. From the bottom of the page, select **Import**.
3. Choose the CSV file to import and select **Save**.

Exporting existing ADP configuration

You can export all of your existing ADP configuration data to a CSV file. This produces a CSV file in the same format as that used for importing configuration data. The file includes a header row.

To export the data:

1. On the Pexip Infinity Administrator interface, go to **Services > Automatically Dialed Participants**.
2. From the bottom of the page, select **Export**. This takes you to the **Export Automatically Dialed Participant Configuration** page.
3. Select **Download**.
4. Follow your browser's prompts to save or open the file.

Best practices

Performing routine checks

Here is a list of tasks that you should perform on a regular basis to ensure that your Pexip Infinity deployment is running optimally.

Check live status

The [Live view](#) page gives a real-time overview of your deployment, and should be viewed regularly. In particular, check for the following:

- Whether there are any alarms on the Management Node or Conferencing Nodes. Any alarms are listed in the top right of the Live view page, and any Conferencing Nodes that have an alarm are shown on the graph with an orange or red outline. For more information, see [Viewing alarms](#).
- The current usage and capacity of your deployment.
- Any conferences or conference participants that are experiencing call quality issues.

For more information, see [Viewing live and historical platform status](#).

Check alarm history

Active alarms are cleared as soon as the issue is resolved, so you may not always be aware of temporary issues such as lack of capacity or licenses. However, the **History & Logs > Alarm History** page shows the details of all alarms that have been raised, so it is useful to review this information to see if there are any trends that need to be addressed.

For more information, see [Viewing alarms](#).

Check LDAP sync status

If you are [Provisioning VMRs, devices and users from Active Directory via LDAP](#) in your deployment, periodically go to **Status > LDAP Sync** and check each configured sync template for any warnings or errors. If there are any, you should investigate, remedy and re-sync. For more information, see [Viewing LDAP sync template results](#) and [Troubleshooting LDAP server connections](#).

Check usage history

For each location, go to the [historical usage graph](#) and review the level of usage. If usage is nearing existing capacity, you may need to consider adding more capacity.

Check for upcoming software releases

Make sure you are aware of the features that are coming in future releases of the Pexip Infinity software, and ensure you have a plan in place for upgrading your current deployment.

- Sign up for the Pexip newsletter at <https://www.pexip.com> (enter your address in the subscription section at the bottom of the page).
- Review what's in the current release at https://docs.pexip.com/admin/whats_new.htm.
- Review whether any fixes and new features are relevant for your environment at https://docs.pexip.com/admin/release_notes.htm.

Perform regular backups

You should take regular backups of the configuration data on your Management Node. Conferencing Nodes do not generally require backups, as they receive all their configuration information from the Management Node.

For more information, see [Backing up and restoring configuration](#).

Advanced checks

The following checks require an in-depth understanding of the Pexip Infinity platform. We recommend that you attend an appropriate training course — for more information, visit the [Pexip Academy](#).

- Query participant history and filter on `tx_packet_loss_gte=4` and `rx_packet_loss_gte=4` to see any calls that have more than 4% packet loss. If so, investigate the cause and consider whether further action is required.
- Search administrator and support logs for `irregular_pulse`. This is usually indicative of over-committed hardware. For more information, see [Hardware instability detected](#).
- Search administrator and support logs for `incident`. If there are any, contact your Pexip authorized support representative. If your deployment is [Automatically reporting errors](#), you may have already been contacted about the incident.
- Search administrator log for `error` and `warning`.

Security best practices

The Pexip Infinity platform uses industry-standard encryption and security protocols to control access and to prevent unwanted audiences from listening in and stealing communications. It is also designed to comply with the strictest US Federal requirements.

This topic describes how you can help secure your Pexip Infinity deployment from network-based attacks and integrate Pexip Infinity into your existing network security architecture. For a full discussion of general best practices to prevent your videoconferencing deployment from being compromised, read this Pexip [white paper](#).

Attacks on the operating system and management interfaces

Pexip Infinity uses a customized, cut-down version of Linux which has been designed to avoid exposing unnecessary network services and thus naturally limits the "attack surface" available to an attacker. Pexip regularly releases new software versions which incorporate the very latest operating system security patches (see [Pexip security bulletins](#) for more information). In addition, all Pexip Infinity APIs and management interfaces are password or PIN protected.

To mitigate attacks on the operating system and management interfaces:

- Disable the local management account and instead authenticate and authorize login accounts via a centrally managed Windows Active Directory / LDAP-accessible server. You can also limit what certain administrator groups can do (for example, support teams may not need the ability to deploy Conferencing Nodes). You can also configure client-certificate-based access if required. See [Managing administrator access via LDAP](#) for more information.
- Configure [Global settings](#) to define an administration session timeout and login banner text.
- Configure [Global settings](#) to disable access over SSH if it is not required.
- If you use Simple Network Management Protocol (SNMP), use SNMPv3 to both encrypt and authenticate incoming SNMP discovery and monitoring between the Management Node and the SNMP manager (see [Monitoring via SNMP](#)).
- Use secure NTP to obtain accurate system time (see [Syncing with NTP servers](#)).
- Use a firewall to prevent unauthorized network traffic from reaching your devices, and to block unauthorized access to services and network ports that are not required to be exposed for video communications to work correctly. For example, the management user interface HTTPS, SNMP and SSH services do not usually need to be accessible to anyone other than your network administrator. See [Pexip Infinity port usage and firewall guidance](#) for more information.
- Use secure remote logging via the industry-standard syslog protocol (see [Using a syslog server](#)).
- Use the latest release of Pexip Infinity software.

DOS/DDOS attacks

Pexip Infinity, as with many network services, can be vulnerable to Denial of Service (DOS) or a Distributed Denial of Service (DDOS) attacks. To mitigate such attacks:

- Use a firewall to block unauthorized access to services and network ports that are not required to be exposed for video communications (see [Pexip Infinity port usage and firewall guidance](#)).
- Disable unneeded services altogether. You can do this by configuring services and protocols via [Global settings](#).

Eavesdropping and rogue calls

Pexip Infinity supports the latest industry standards for encryption for communication with end-user devices and employs IPsec security to provide strong protection of all inter-node communications (see [Encryption methodologies](#)). Inter-node traffic is restricted to only protocols that are expected; any unexpected traffic/protocols are dropped. Pexip Infinity also works with all popular video call control systems in the market today, and can connect legacy devices in the corporate network (which may not themselves support encryption) and encrypt on behalf of those devices when connecting to external devices that do support encryption.

Common attacks on videoconferencing systems include rogue calls — such as Spam Over Internet Telephony (SPIT) or toll fraud call attempts — that are targeted at an organization's SIP (or, more rarely, H.323) infrastructure. Typically the attacker will place a large volume of calls to numeric aliases (usually using SIP UDP) to try and gain access to a VoIP to PSTN gateway — and, if successful, use the gateway to commit toll fraud.

To mitigate eavesdropping and rogue calls:

- Use proper TLS/SSL certificates from a respectable source on all Pexip Infinity nodes so that clients and other servers can verify that they have genuinely connected to the correct Pexip Infinity server and not an impostor or "man-in-the-middle" (see [Managing TLS and trusted CA certificates](#)). You can also:
 - Use OCSP to check the status of certificates.
 - If appropriate, enable SIP TLS verification / mutual authentication to require that peer certificates are verified. Typically, this is recommended for Microsoft Skype for Business and Lync integrations, but most other videoconferencing deployments are **not** equipped to provide a proper SIP TLS certificate so SIP TLS verification is not recommended unless you only expect calls from a closed circle of users.

Note that all Pexip Infinity nodes use HSTS (HTTP Strict Transport Security) to ensure greater security.

- Enable PIN protection on your Virtual Meeting Rooms:
 - use a long (at least 6 digits), unique, randomly-generated PIN for each Virtual Meeting Room (you can automate this if you are [Provisioning VMRs, devices and users from Active Directory via LDAP](#)),
 - [use a trailing #](#) at the end of each PIN to disguise the length,
 - regularly change the PIN on each VMR.
- When using numeric aliases for your VMRs, make them at least 6 digits long.
- If you are using the Pexip Reverse Proxy and TURN Server, you should enable fail2ban on the reverse proxy.
- Ensure that **PIN brute force resistance** and **VOIP scanner resistance** are enabled, either [globally](#) or in specific [locations](#).
- When using the Infinity Gateway, consider:
 - Limiting your Call Routing Rules applicability to only allow calls to be made from devices that are registered to Pexip Infinity, or are received in certain locations.
 - Limiting calls to certain incoming protocols e.g. SIP only.
 - Only allowing calls to be placed to devices that are registered to Pexip Infinity.
 - Using precise regular expressions in your Call Routing Rules for the domains, dial plans and alias patterns that you want to support.

See [Configuring Call Routing Rules](#) for more information about routing calls via the Infinity Gateway service.

- Pexip Infinity disables SIP UDP by default (as SIP UDP is the most commonly targeted signaling service). However, consider disabling other unused protocols (see [Enabling and disabling SIP, H.323, WebRTC and RTMP](#)).
- Disable any other features that are not required in your deployment, such as the ability to make outbound calls, and support for the Infinity Connect / API clients (see [Global settings](#)).
- Protect against toll fraud by ensuring that access to your VoIP gateway or your VoIP provider's SIP trunk and other important resources is carefully restricted – especially for unauthenticated external SIP/H.323 callers (see [PSTN gateways and toll fraud](#)).
- Implement a local policy script such as our example Reject specified user agents script that rejects calls from known bad User Agents. This script will deflect the majority of "casual" scan attacks.
- Monitor the administrator log and use features of your firewall to block offenders.

Resilience strategies — redundancy, backing up and restoring data

This topic discusses the resilience, redundancy, backup and restore strategies to consider when deploying and maintaining your Pexip Infinity platform.

Resilience and redundancy

Pexip is designed for multiple layers of resilience and redundancy. Companies and service providers should consider which situations they want to protect against. All options can be combined, and this is typically a consideration of cost versus benefit, and how much downtime can be tolerated in a worst case scenario.

The main levels of resilience redundancy and our associated recommendations are described below.

Hardware and physical considerations

- Dual hot swap power supplies for each server connected to different power circuits, optionally with UPS or backup power.
- Dual network cards in each server, connected to dual switches (VMware NIC Teaming). Switches are connected to redundant routers, allowing for any component in the network path to fail. Consider if the data center is robust if the fiber cable to the data center is cut.
- Redundant storage, either by adding a hardware RAID controller and operate two disks in RAID 1 (mirror) or by using redundant external SAN solutions.
- Redundant servers — we recommend that service providers deploy n+1 to always allow for one physical server to be unavailable.
- Redundant datacenters — consider providing Pexip Infinity from multiple data centers (either multiple data centers in one region or in various international regions).

VMware considerations

Note that some of the options listed below are not relevant for all node types in a Pexip deployment. Typically a Pexip Management Node, Reverse Proxy and the Conferencing Nodes responsible for signaling should be kept up by using technologies such as vMotion, DRS, or High Availability.

- For Conferencing Nodes with a consistently high media load, we recommend deploying n+1 (or more) Conferencing Nodes, so that you always have spare capacity if a node becomes unavailable. If using High Availability for Conferencing Nodes, configure them with resource reservation to ensure that they are not starved for resources if they are moved. Also ensure that Conferencing Nodes are only moved between servers with equal core count per CPU to avoid NUMA issues.
- vMotion: proactively move the node to another server if maintenance/changes are to be conducted on the server where it is currently running. Do not vMotion a Conferencing Node that has active conferences.
- DRS: automatically move VMs around depending on load for servers. This should not be used with Conferencing Nodes that handle media.
- High Availability: this is VMware's ability to automatically start a VM on a different host, if the main server for this VM is not operational. This requires good understanding of VMware to handle this correctly, and requires ensuring that the VM boots up on a server with appropriate resources.
 - Recommended for the Management Node, Proxying Edge Nodes and the reverse proxy, to ensure they are brought to life in case of a server failure.
 - Should only be used for Transcoding Conferencing Nodes after careful planning.

Call control or DNS considerations

- If using Pexip Infinity to provide registration and call control services, deploy multiple Conferencing Nodes to provide resiliency options.
- If using DNS SRV records for call control server discovery, ensure that each SRV record points to multiple A records (the names of multiple servers) so that call control can still be provided by other Conferencing Nodes if one fails.
- Consider DNS SRV records with different priorities that fail over to an alternative datacenter if available.
- Be prepared to modify DNS configurations to direct traffic to alternative datacenters in case of a major disaster in one datacenter.
- Ensure that call control systems outside of Pexip (other SIP registrars, H.323 gatekeepers, Skype for Business servers, or other components) are integrated with multiple Pexip Conferencing Nodes (often via the DNS strategies mentioned above) to allow service continuity should any node in the Pexip deployment be unavailable.

See [DNS record examples](#) for more information.

Physical locations of Pexip nodes

As the Pexip Infinity platform can be deployed with multiple nodes in multiple locations, make use of this flexibility and design the setup as required for the spread of customers being served.

Backup mechanisms

It is mandatory for any critical installation to have a proper backup of the Management Node.

The Pexip Infinity platform can be backed up in many ways — with different advantages to each option. In some scenarios, for fast recovery of various situations, using multiple (or all) options is possible, and in most cases we recommended using at least two backup strategies.

For on-premises deployments, we recommend that you use both the hypervisor and Pexip's inbuilt methods to preserve your configuration data. A VM snapshot should be your primary mechanism prior to an upgrade, as this allows you to easily restore your system back to its state at the time the snapshot was taken. The Pexip Infinity backup and restore mechanism is your fallback mechanism, as this allows you to preserve a copy of your data in an alternative location, in case you lose your VM environment. Cloud-based deployments (Azure, AWS, GCP or Oracle) should use the Pexip Infinity backup and restore mechanism.

- i** In all cases ensure that you take regular backups, refreshing your backups after making configuration changes to your Management Node.

VMware backup

VMware backups should be performed in the same datacenter (with an offsite replication of the backup in case of a local disaster) to allow fast restoration of the data.

Be aware that snapshots are not backups. Snapshots are a tool to roll back to a given time. Therefore, we recommend taking snapshots only when necessary (such as prior to an upgrade) and deleting the snapshot as soon as possible after the upgrade is confirmed to be successful. You should only create and delete VMware snapshots at a time of minimal usage. Taking or removing snapshots can cause virtual machines to become unresponsive.

VM backups should use a proper hypervisor VM backup tool (e.g. VMware VDP — vSphere Data Protection) or similar, and restoration should be tested and verified (preferably after the inbuilt backup methods have been set up, to ensure that you have another way of recovering if your restoration fails).

Conferencing Nodes do not need to be backed up. They receive all their configuration information from the Management Node and can simply be redeployed if necessary. However, if your Conferencing Nodes are geographically spread out and redeploying them would consume significant bandwidth or take a significant length of time, they can also be backed up with your hypervisor's backup tools.

Pexip's inbuilt Management Node configuration backup process

You can use Pexip Infinity's inbuilt backup and restore mechanism to backup and restore the configuration data on the Management Node.

You can enable daily automatic backups, and you can also take a manual backup whenever it is appropriate, for example, before and after you make any configuration changes or perform a software upgrade.

- All backup files are encrypted — the administrator supplies a passphrase and must remember this for any subsequent restoration.
- Restoration must occur on exactly the same version that the backup was taken from.
- The data contained in the backup contains all configuration data, including IP addresses, custom themes, certificates and call history.
- The backup data does not contain licenses, the administrator log, the support log, usage statistics or the operating system password.
- The system keeps on the host VM only the 5 most recent manually-taken backups and the 5 most recent automatic backups. Older backup files are deleted.

As the restore does not contain the license key, if you use this method to restore your configuration data onto a fresh Management Node you will need to contact your Pexip authorized support representative to obtain a new license key.

We recommended that you configure Pexip Infinity to schedule regular backups and send the backup file to an external FTP server. See [Backing up and restoring configuration](#) for instructions.

Pexip import/export of service configuration data

This is your "last resort" if you do not have the ability to restore configuration from a VM backup or from a Pexip backup. You can export and import your service configuration data (Virtual Meeting Rooms, Virtual Auditoriums, Virtual Receptions, device aliases and Automatically Dialed Participants) to and from CSV files.

You can apply this data to a fresh (or existing) Pexip installation if, for example, you had to redeploy the entire Pexip platform. Note that Call Routing Rule and other platform configuration must be documented for manual restoration in this scenario. See [Bulk import/export of service configuration data](#) for instructions.

If you use [provisioning](#) you can restore your VMRs and device aliases from your LDAP/AD source.

Impact of lost connectivity to the Management Node

If the Management Node disappears from the network (due to network outage, server outage, lack of VMware HA, or during HA restoration etc.) there will be some impact on your Pexip Infinity platform:

- **No configuration updates pushed to Conferencing Nodes:** a Conferencing Node typically checks in with the Management Node once every 60 seconds for configuration updates. This will fail, hence no new VMRs, Call Routing Rules, or platform configuration updates such as DNS servers etc. will be added to the Conferencing Node's local replicated database.
- **No CDR/log data sent back to the Management Node:** the Conferencing Nodes will try to push syslogs back to the Management Node with log data. This will also fail, and the Conferencing Nodes will buffer the log data until the Management Node is operational (there may be some limitations here — if you leave the Management Node down for a very long time and there is a lot of traffic, at some point the Conferencing Node logs will rotate to avoid filling up the disk, but you should not normally have such long outages).
- **No visibility in Pexip Management Node interface:** new calls will not appear in the Management Node's Administrator interface (assuming the Administrator interface is accessible, but some Conferencing Nodes cannot reach the Management Node due to a network split, for example).
- **Licensing:** if a Conferencing Node is unable to contact the Management Node, the call is permitted on the assumption that a license is available. After the Management Node has been out of contact for a grace period of 14 days, all calls will be rejected.
- **Rebooting/restarting Conferencing Nodes:** do **not** restart or reboot a Conferencing Node while the Management Node is unavailable. The 14-day licensing grace period only works if the Conferencing Node has had a valid sync with the Management Node after the Conferencing Node's most recent reboot.
- **Syslog to external syslog server (TCP/TLS):** for reliable syslog output, each Conferencing Node can send its syslog data directly your own external syslog server, for quick realtime analysis with tools like Splunk etc. By using TCP/TLS based syslog you will use a reliable data channel so you know that the traffic is received in the other end. As the log data is sent directly from the Conferencing Nodes to your configured syslog server, it is not affected by the Management Node not being available.

Some features are not affected or are only partly affected:

- **New and existing calls:** Conferencing Nodes will continue to handle both new and existing calls as before — this is because no conference media or signaling passes through the Management Node.

Considerations when restoring the Management Node from backup

Your procedures to restore the Management Node from a backup depend upon the type of backup available and the problem that has occurred:

Restoring from VMware

This is probably the quickest and most reliable way of restoring the Management Node.

- You can perform a Management Node restore from VMware during production hours.
- When the restore is complete the configuration replication will start as soon as the IPSec tunnels between the Management Node and the Conferencing Nodes are back up. Due to the nature of IPSec, some Conferencing Nodes might take a while to sync back up. Rebooting a Conferencing Node will solve this, but that will impact operations and will drop any active calls that are being handled on that node — and note that if the node is not in sync with the Management Node you cannot use the Administrator interface to check if that node has any running calls.
- Any configuration on the Conferencing Nodes will be refreshed with the data that has been restored to the Management Node.
- Any logs / call data records (CDRs) that have already been sent to the old Management Node will be lost as Conferencing Nodes do not check what is in the Management Node database — it only knows they had been delivered. To recover these CDRs, use external syslog data and parse it as an additional import to your billing system. New CDRs will be queued awaiting the Management Node to become operational again.

Restoring a file created by Pexip's inbuilt Management Node configuration backup process

We recommend using this approach if the Management Node needs to be reinstalled or rebuilt.

When you perform a restore of Management Node configuration data, the node's IP address is restored (as this is used by the IPSec tunnels to all of the Conferencing Nodes). To be able to restore a Management Node into another datacenter (in case of local disaster

recovery), the Management Node needs an IP address from your ISP that can be re-routed to another site. For high-availability deployments, this is something worth discussing with the network provider as you cannot just move nodes around after being deployed.

- You can perform a Management Node restore from a Pexip backup during production hours.
- If you are restoring onto a fresh Management Node:
 - As the IP address of the new Management Node will be the same as the old node, they should not both be powered on in the same VLAN to avoid IP conflict.
 - It will not replicate configuration to the Conferencing Nodes until the backup file is restored (as the new Management Node does not contain the right certificates to communicate with the Conferencing Nodes). When the Pexip backup is restored, it will connect with the Conferencing Nodes and replicate configuration. Any configuration on the Conferencing Nodes will be refreshed with the data that has been restored to the Management Node.
 - Any logs / call data records (CDRs) that have already been sent to the old Management Node will be lost as Conferencing Nodes do not check what is in the Management Node database — it only knows they had been delivered. To recover these CDRs, use external syslog data and parse it as an additional import to your billing system. New CDRs will be queued awaiting the Management Node to become operational again.

Restoring service configuration data from CSV files

An import of service configuration data can be performed at any time.

- The primary key for the Virtual Meeting Room / Virtual Auditorium / Virtual Reception database is the name of that service. So, for example, if you import VMRs on top of an existing VMR database, the existing configuration for those VMR names will be overwritten by the imported VMR's configuration. Any other existing VMR records will be left unchanged.
- The primary key of device alias data is the alias itself, and for Automatically Dialed Participants it is the combination of the conference name and the alias to be dialed.
- Any new or modified service configuration will take effect after approximately 60 seconds.
- CDR data is not affected by a restore of service configuration data.

PSTN gateways and toll fraud

If your environment includes a PSTN gateway or uses an ITSP (Internet telephony service provider), consider the potential for toll fraud if you have Call Routing Rules that can route calls to the PSTN gateway or ITSP, or if you allow conference participants to dial out to other participants via the PSTN gateway or ITSP.

As you might intentionally want to allow users to route calls via the PSTN gateway or ITSP (and thus incur toll charges), we recommend that you use a suitable call control solution such as a Cisco VCS to configure an appropriate dial plan and authentication mechanism for your network.

The ways in which Pexip Infinity calls may be routed via a PSTN gateway or ITSP are described below.

Infinity Gateway calls

If your dial plan allows Infinity Gateway calls and has a Call Routing Rule which, for example, matched 9.* and routed the call via the PSTN gateway or ITSP, this would allow anyone who could route a call to the Pexip Infinity platform to then send a call via the PSTN gateway or ITSP.

You could restrict those rules to only apply to incoming calls from registered devices, or to calls that are being handled in an internal location.

Virtual Receptions

If you have configured Virtual Receptions and also have Call Routing Rules that match numeric aliases (such as 9.*), then anyone that can reach the Virtual Reception could match the Call Routing Rule and potentially route their call via the PSTN gateway or ITSP.

Thus, if any call control rules are in place to restrict who may dial numbers which correspond with numeric Call Routing Rules, then the same restrictions should also be placed on who may call any Virtual Receptions.

Manually dialing out to a participant from a conference

Pexip Infinity allows you to manually dial out to participants from a conference, on an ad hoc basis.

This means that conference participants using Infinity Connect clients could place outbound calls via the PSTN gateway or ITSP. Note that these types of calls may dial out directly to the destination alias or they may use Call Routing Rules.

In these circumstances, to reduce (but not eliminate) the risk of toll fraud, we recommend that you use PIN-protected conferences (so that only Hosts can dial out).

Example emails for sending to new users

Based on feedback from our customers, we know that issues can occur when end users who are new to Pexip Infinity don't understand what clients to use, or don't know what aliases to dial in order to join a meeting. So we've put together some best practice templates that you as an administrator can use as a basis for emails you'll send to your users, to let them know how they can access their Virtual Meeting Rooms and let others join them for meetings, and which Infinity Connect clients to use and how to use them.

All deployments will differ, but some general recommendations and considerations are:

- Your email to new users should be very clear about how they can access their own VMR, and also provide clear instructions for them to forward on to anyone they want to invite to their VMR as a guest. You should also make it clear which devices or clients they should use when calling in to your deployment.
- Users connecting from outside your organization and who do not have their own video device should generally use the Infinity Connect web app to access VMRs. This means that they won't need to download or install anything in order to access meetings, but will still have the same high-quality user experience and functionality of participants using the Infinity Connect desktop client. You'll need to make sure that at least one Conferencing Node is accessible externally, and you'll also need to [set up appropriate DNS records](#) for connections from both inside and outside your network.
- Users connecting from inside your organization should also use the Infinity Connect web app, unless you want them to be able to register to receive incoming calls — in which case they need to use the Infinity Connect desktop client.
- The Infinity Connect desktop client should be used if you want to take advantage of the additional registration (to receive incoming calls) and internal directory service features. Administrators can also set up [Call Routing Rules](#) that apply to registered devices only, meaning that you can permit registered Infinity Connect desktop client users to make calls that Infinity Connect web app users cannot.

If you are deploying the Infinity Connect desktop client in your environment, we recommend that you make use of [provisioning](#), and you'll also need to [set up appropriate DNS records](#).

- If you have a Lync / Skype for Business environment, consider whether you want users to continue to use those familiar clients to access their VMRs.
- Consider whether you want users to be able to join or control meetings from a mobile device. If so you'll need to [set up appropriate DNS records](#).
- Consider whether you want to allow users to call in to meetings from a telephone. If so, you will need to [set up appropriate telephone numbers](#) and Virtual Receptions.
- Where possible, simplify the join experience for users by [providing URLs](#) that they can click to join meetings directly.

We already provide example [joining instructions](#) for use with the VMR Scheduling for Exchange feature, and provisioning, but here we also give some examples for [Accessing a VMR](#) and [Using Outlook to schedule meetings in a VMR](#).

Accessing a VMR

Below is some example text you can send to new users to let them know how they and their guests can access their VMRs. You should edit this as appropriate for your own deployment and for each individual user by:

- adding the URLs
- specifying the VMR aliases and Host PINs for each user's VMR
- specifying the Guest PIN (or deleting references to Guests PINs if you don't use these in your deployment)
- adding download links for the Infinity Connect desktop client
- deleting any content not relevant to your deployment
- adding formatting and hyperlinks to the text.

You can include URLs in the text that, when clicked, will open a specific client and pre-fill some of the required fields, such as VMR alias and PIN. For guidance on creating these URLs, see [Creating preconfigured links to launch conferences via Infinity Connect](#).

Welcome to your own personal Virtual Meeting Room (VMR).

You will usually be the host for any meetings in this VMR. This means you can control when the meeting starts and ends, and you can also mute and unmute noisy participants, lock the virtual meeting room (to prevent anyone else from joining part way through the meeting), and eject unwanted participants.

How to join your VMR as a host

- Using a web browser from within <organization>, go to <internal URL> then enter your Name and select "Connect".
- From a web browser outside <organization>, go to <external URL> then enter your Name and select "Connect".
- From a video conferencing endpoint inside <organization> dial <internal alias> then enter the PIN: <Host PIN>.
- From a video conferencing endpoint outside <organization> dial <external alias> then enter the PIN: <Host PIN>.
- From Skype for Business, dial <alias> then enter the PIN: <Host PIN>.
- From an Infinity Connect desktop client, first download the client from <link>. After it is installed, click this link <URL>, or dial <alias> then enter the PIN: <Host PIN>.
- From a telephone, dial <telephone no.> then, when prompted, enter the conference number <numeric alias> followed by "#". When asked, enter the host PIN: <Host PIN>.
- To control the meeting from an Android or iOS device, first install the client from the Google Play store or the Apple store. After it is installed, click this link <URL> or dial <alias> then enter the PIN: <Host PIN>.

How your guests can join your VMR

You can copy and paste the text below when inviting guests to join your VMR.

To join my VMR, you can simply use a web browser if you don't already have a video device you can use:

- Using a web browser from within <organization>, go to <internal URL> then enter your Name and select "Connect".
- Using a web browser from outside <organization>, go to <external URL> then enter your Name and select "Connect".
- From a video conferencing endpoint inside <organization> dial <internal alias> then enter the PIN: <Guest PIN>.
- From a video conferencing endpoint outside <organization> dial <external alias> then enter the PIN: <Guest PIN>.
- From Skype for Business, dial <alias> then enter the PIN: <Guest PIN>.
- From an Infinity Connect desktop client within <organization>, click this link <URL>, or dial <alias> then enter the PIN: <Guest PIN>.
- From a telephone, dial <telephone no.> then, when prompted, enter the conference number <numeric alias> followed by "#". When asked, enter the host PIN: <Host PIN>.

If you arrive before the meeting host, you will be placed in a virtual waiting room. You will be automatically admitted to the meeting when the host arrives.

Using Outlook to schedule meetings in a VMR

Below is some example text you can send to new users to let them know about the VMR Scheduling for Exchange feature. An end user guide for this feature is available (see [Using Outlook to schedule meetings in VMRs](#)); you can either copy some information from there, or send your users the direct link to the guide.

The [way in which users locate and activate the add-in](#) will depend on the Outlook client being used, so you will need to replace <Pexip Scheduling Service option> with the relevant details for your deployment.

Using Outlook to schedule video meetings

As part of the roll-out of Virtual Meeting Rooms (VMRs) in our organization, you now have the ability to schedule any of your meetings as video meetings, using a one-off VMR created just for that meeting.

To do this, just create your meeting invitation in the usual way and then select the <Pexip Scheduling Service option>.

This will automatically add joining instructions to the body of the meeting invitation.

Note that for these sorts of meetings, everyone joins in the same way using the same links, and everyone has host privileges.

For full instructions, see www.docs.pexip.com/admin/scheduling_user_guide.htm

Pexip Infinity reference information

Glossary of Pexip Infinity terms	423
Regular expression (regex) reference	428
Jinja2 templates and filters	432
Encryption methodologies	435
Interoperability	436
Supported RFCs	437
Patents	439
Accessibility	440
Using Microsoft Skype for Business / Lync with Pexip Infinity	441
When is a reverse proxy, TURN server or STUN server required?	443
Integrating with streaming and recording services	444
Integrating with telephone systems (PSTN)	467

Glossary of Pexip Infinity terms

The following table contains a list of many of the terms used in Pexip Infinity documentation.

Term	Definition
Adaptive Composition	An intelligent layout with real-time automatic face detection and framing that is used in VMRs.
Alarm	An alert that is raised when there is an issue on your Pexip Infinity deployment that requires attention. For more information, see Viewing alarms .
Alias	<p>The string that, when received by a Conferencing Node, triggers the creation of a conference instance (or if one already exists, causes the call to be routed to the appropriate conference).</p> <p>Each alias is associated with a conferencing service (Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service), which defines the type and settings (such as PIN) for the conference that is created.</p> <p>In most cases, the alias received by the Conferencing Node will be the same as the alias that the conference participant dialed from their endpoint, but there are some exceptions (for more information see About aliases and access numbers).</p> <p>Depending on the dial plan, multiple aliases can be used throughout a network to access the same service.</p>
Automatically dialed participant	<p>A participant/device that will have a call placed to it from a Virtual Meeting Room or Virtual Auditorium whenever a conference using that service starts.</p> <p>For more information, see Automatically dialing out to a participant from a conference.</p>
Backplane	<p>A link between the Management Node and a Conferencing Node, or between two Conferencing Nodes, used to transmit Pexip control messages. Backplanes between Conferencing Nodes also transmit conference audio, video, and data packets.</p> <p>All packets are secured through authentication and encryption designed to protect the privacy of the data. For more information, see Encryption methodologies.</p> <p>Local backplanes exist between Conferencing Nodes in a single location.</p> <p>Geo backplanes exist between Conferencing Nodes in different locations.</p>
Bursting	<p>Pexip Infinity deployments can burst into cloud-hosted services when primary conferencing capabilities are reaching their capacity limits, thus providing additional temporary Transcoding Conferencing Node resources. See Dynamic bursting to a cloud service for more information.</p>
Call tag	<p>An optional identifier for each participant in a call that can be specified in client API requests and then used by app developers to correlate other API requests. For more information, see Tracking usage via service and participant call tags.</p>
Cloud service	<p>As an alternative to deploying your Pexip Infinity platform on premises (in your own datacenters), you can deploy nodes on a hosted cloud platform. See Deploying as a cloud service via Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP) or Oracle Cloud Infrastructure for more information.</p>
Conference	<p>A general term that can be used to refer to either:</p> <ul style="list-style-type: none"> a configurable service (such as a Virtual Meeting Room or Virtual Auditorium) that can be used to host virtual meetings, or a specific instance of a meeting being held in a Virtual Meeting Room or Virtual Auditorium.
Conference instance	<p>A conference with active participants that exists on one or more Conferencing Nodes. A unique conference instance is created when the first participant dials an alias associated with a Pexip Infinity service alias, and lasts until the last participant disconnects.</p>

Term	Definition
Conferencing Node	<p>A virtual machine (VM) that provides the capacity for conferences, handling the media processing and call signaling.</p> <p>A Conferencing Node can have either a transcoding or a proxying role:</p> <ul style="list-style-type: none"> • Transcoding Conferencing Nodes are required in all deployments; they manage all of the media processing required to host a conference, and can also handle direct connections to/from endpoints if required. • Proxying Edge Nodes are optional; they handle call signaling and the media connection with the endpoint, but forward the media on to a Transcoding Conferencing Node for processing. <p>For more information, see Conferencing Nodes.</p>
Connection (to a conference)	<p>A connection can be a call or presentation from an endpoint to a Virtual Meeting Room or Virtual Auditorium, a backplane between Transcoding Conferencing Nodes, or a call into or out of the Infinity Gateway. In this context, a connection is analogous to a port. In some situations, a single conference participant such as a WebRTC or Skype for Business client requires two connections (one for the video call, and one for presentation content).</p>
Core	<p>One single physical processing unit. An Intel Xeon Scalable processor typically has between 8 and 32 cores, although both larger and smaller variants are available.</p>
CVI	<p>Cloud Video Interop (CVI) is a Microsoft Qualified third-party solution that enables third-party meeting rooms (telepresence) and personal video devices (VTCs) to join Microsoft Teams meetings. Pexip Infinity is a Microsoft-certified video interoperability platform for Microsoft Teams. See Integrating Microsoft Teams with Pexip Infinity.</p>
Distributed conference	<p>A conference instance that exists across two or more Conferencing Nodes. It can be locally distributed, globally distributed, or both:</p> <p>Locally distributed conferences exist across two or more Conferencing Nodes in the same location.</p> <p>Globally distributed conferences exist across two or more Conferencing Nodes in physically different locations.</p> <p>Locally and globally distributed conferences exist across two or more Conferencing Nodes in one location and at least one other Conferencing Node in a different location.</p>
Endpoint	<p>A device capable of participating in a conference. The endpoint's capabilities can vary from audio-only to full audio, video, and data sharing support.</p>
Event sink	<p>An external service to which Pexip Infinity can send details of every participant and conference management event. For more information, see Using event sinks to monitor conference and participant status.</p>
Host participant	<p>A conference participant who has privileges to control aspects of the conference. For more information, see About PINs, Hosts and Guests.</p>
Host server	<p>The physical hardware on which the virtual Management Node and Conferencing Nodes reside. For more information, see Host servers.</p>
Hybrid SfB deployment	<p>A deployment of on-premises Microsoft Skype for Business / Lync and Office 365 where SfB/Lync users may be homed in either environment.</p>
Hypervisor	<p>An application that is used to create and manage virtual machines. For more information on the hypervisors supported by Pexip Infinity, see Supported hypervisors.</p>
Infinity Connect	<p>A suite of software clients that allows end users to connect to Pexip Infinity services from a web browser, an installable desktop client, or a mobile client. For more information, see Introduction to Infinity Connect.</p>
Intermediary node	<p>In a globally distributed conference, one Conferencing Node in each location acts as the intermediary for any other Conferencing Nodes in the same location. Each intermediary node funnels traffic between locations. This prevents full mesh bandwidth occupation, minimizing bandwidth consumption and ensuring optimal WAN utilization.</p>

Term	Definition
IVR (Interactive Voice Response)	IVR technology allows participants to use a DTMF keypad to interact with Pexip Infinity services to: <ul style="list-style-type: none"> select which Virtual Meeting Room or Virtual Auditorium they wish to join. For more information, see About the Virtual Reception IVR service. enter a PIN, for those Virtual Meeting Rooms that have restricted access. For more information, see About PINs, Hosts and Guests.
Management Node	A virtual machine (VM) on which the Pexip Infinity software is installed. This machine hosts the Pexip Infinity Administrator interface. It is used to create one or more Conferencing Nodes and configure information about the conferences that can exist on those Conferencing Nodes.
MTU (Maximum Transmission Unit)	The size of the largest packet of data that can be transmitted via the network interface of a device (endpoint, Conferencing Node or Management Node).
NUMA node	The combination of a processor (consisting of one or more cores) and its associated memory.
OAuth 2.0	An authorization framework that enables users to grant third-party access to an HTTP service / web resource without sharing their passwords.
One-Touch Join	Pexip's One-Touch Join (OTJ) allows users to schedule a meeting in Microsoft Outlook or Google Calendar and include in the invitation a meeting room with a supported Cisco or Poly videoconferencing endpoint, so that the endpoint in the chosen meeting room displays a Join button just before the meeting is scheduled to begin. Participants can then simply walk into the room and select the button, and the endpoint will automatically dial in to the meeting.
Participant	A conference participant typically refers to the endpoint device or system that is connected to a conference — this could be a personal device or a room system (where the room might contain multiple people). A participant could be a video participant or an audio-only participant. Such participants are only counted once regardless of whether or not there is a presentation stream in addition to the main video, and they only consume a single call license. An Infinity Connect client can also join a conference as a presentation and control-only participant.
Pexip Infinity	A virtualized and distributed multipoint conferencing platform that comprises a single Management Node and one or more Conferencing Node(s) .
Pexip Infinity software	The files that are used to deploy the Management Node and Conferencing Nodes onto the physical virtual machine (VM) infrastructure or cloud service.
Pexip Smart Scale	The Pexip Smart Scale (PSS) feature allows you to have Conferencing Nodes that are deployed by Pexip on your behalf within the secure Pexip Private Cloud, in the form of Pexip Smart Scale locations . You can add or remove these locations, and scale their capacity up or down, according to your own deployment's changing requirements. For more information, see Enabling Pexip Smart Scale .
Port	<p>The term "port" may be used to describe a connection (such as a call or presentation) from an endpoint to a Conferencing Node or a backplane between Transcoding Conferencing Nodes. For more information, see Pexip Infinity license installation and usage.</p> <p>Port can also refer to the virtual data connections used for network traffic between devices. For more information on the ports used by the Management Node and Conferencing Nodes to connect to other devices, see Pexip Infinity port usage and firewall guidance.</p>
Processor	The hardware within a computer that carries out the basic computing functions. Can consist of multiple cores.
Proxying Edge Node	A Proxying Edge Node that is deployed as a front for internal Transcoding Conferencing Nodes as part of a distributed Pexip Infinity system. For more information, see Distributed Proxying Edge Nodes .
RAM	Also referred to as "memory". The hardware that stores data which is accessed by the processor core while executing programs.

Term	Definition
Reverse proxy	A reverse proxy is an application that can proxy HTTP and HTTPS traffic from an externally-located client to a web service application located on the internal network — in our case a Pexip Conferencing Node. A reverse proxy can also be referred to as a load balancer.
Scheduled conference	A conference held in a Virtual Meeting Room that has been created using the VMR Scheduling for Exchange service. Users can host their meeting in a single-use VMR that is created specifically for the meeting and only available for its duration, or they can host their meeting in their own personal VMR. For more information, see VMR Scheduling for Exchange .
Service	In the context of this documentation, a Pexip Infinity service is one of the following: Virtual Meeting Room, Virtual Auditorium, Virtual Reception, scheduled conference, or Test Call Service. Note that, separate to the Pexip Infinity self-hosted platform described in this documentation, Pexip also has a managed video conferencing-as-a-service (VCaaS) offering, referred to as the Pexip Service .
Service tag	An optional identifier that an administrator can assign to a service, allowing them to track usage of the service via the administrator log. For more information, see Tracking usage via service and participant call tags .
Socket (CPU)	The socket on the host server's motherboard where one processor is installed.
System location	A label that allows you to group Conferencing Nodes together, typically according to where they are physically located. For more information, see About system locations .
Teams Connector	The Pexip Teams Connector is a Pexip application that is deployed in Microsoft Azure and is used to enable Microsoft Teams Cloud Video Interop (CVI). It handles all Teams communications and meeting requests from the Pexip Infinity platform and passes them on to the Microsoft Teams environment.
Test Call Service	A Pexip Infinity loopback service that allows users to check the quality of their video and audio prior to joining a conference, and verifies that they can connect to a Conferencing Node.
Thumbnail	A smaller window at the bottom of the main picture which displays the live video stream from a conference participant. Sometimes referred to as a PiP (Picture in Picture).
Transcoding Conferencing Node	In large Pexip Infinity deployments, a Transcoding Conferencing Node handles all the media processing, protocol interworking, mixing and so on that is required in hosting Pexip Infinity calls and conferences. When combined with Proxying Edge Nodes, a transcoding node typically only processes the media forwarded on to it by those proxying nodes and has no direct connection with endpoints or external devices. However, a transcoding node can still receive and process the signaling and media directly from an endpoint or external device if required.
TURN server	A TURN server is a media relay/proxy that allows peers to exchange UDP or TCP media traffic whenever one or both parties are behind NAT.
Virtual Machine (VM)	A software implementation of a computer, which runs on a host server and is implemented and managed using a hypervisor. The Management Node and Conferencing Nodes are virtual machines. In a cloud environment it is typically referred to as a VM instance.
VM manager	An application that allows you to connect to one or more VMware vSphere ESXi Hypervisors (which manage host servers and their virtual machines). VM managers were used in versions prior to v26 to enable automatic deployment of Conferencing Nodes.
Virtual Auditorium	A meeting space that is optimized for use by a small number of Host participants and a large number of Guests. For more information, see About Pexip Infinity conferences .
Virtual Meeting Room (VMR)	A personal virtual meeting space. For more information, see About Pexip Infinity conferences .
Virtual Reception	A central IVR service from which participants can select the Virtual Meeting Room or Virtual Auditorium they wish to join. It also can be used to route a call into an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet. For more information, see About the Virtual Reception IVR service .

Term	Definition
VMR Scheduling for Exchange	A licensed feature within Pexip Infinity that enables Microsoft Outlook users to schedule meetings using Pexip VMRs as a meeting resource. For more information, see VMR Scheduling for Exchange .
VMR self-service portal	The Pexip VMR self-service portal is a separately-installable component that allows end-users to manage their personal Virtual Meeting Room without having to send requests to their administrator to change the configuration of their VMR.

Regular expression (regex) reference

Regular expressions can be used in conjunction with Pexip Infinity features including [Call Routing Rules](#), [Provisioning VMRs, devices and users from Active Directory via LDAP](#), configuring [Regex-based meeting processing rules](#) for One-Touch Join, and searching the [support](#) and [admin](#) logs.

This topic contains information on:

- [Regex testing tool](#)
- [Regex syntax](#)
- [Pattern matching examples](#)
- [Search and replace examples](#)

Regex testing tool

The Pexip Infinity Administrator interface contains an inbuilt regex testing tool:

1. Go to **Utilities > Regular Expression Tester**.
2. Enter your test input and regex:

Option	Description
Input	The test input to match against the regular expression, such as a dialed alias.
Regex match	The regular expression that the Input is checked against (see Regex syntax below).
Regex replace string	The optional regular expression string used to transform the Input (if a match was found). Leave this field blank to leave the original input unchanged.

3. Select **Test Regular Expression** (at the bottom of the page).

The **Result** field shows the result of transforming the **Input** using the **Regex match** and **Regex replace string**. If a **Regex replace string** was not provided then the **Result** will be the same as the **Input**, providing it was a full match.

The **Debug information** shows extra information such as whether it was a successful, partial or failed match, and lists any match groups.

Regex syntax

BRE (basic regular expression) syntax is used when searching the support and admin logs.

Otherwise (i.e. for call routing and provisioning), Pexip Infinity supports case-insensitive Perl-style regular expression patterns as described in the rest of this topic. The table below describes some of the special characters that are commonly used in regular expressions, and provides examples of how they can be used.

Character	Description	Example
Basic syntax		
non-special character	Matches the specified characters literally (providing they are not any of the regex special characters).	abc matches abc, ABC, aBC etc
(...)	Groups a set of matching characters together. Multiple groups can be specified. Each group can then be referenced in order (from left to right) using the characters \1, \2, etc. as part of a replace string.	See Search and replace examples below
\	Escapes a regular expression special character: {}[]()^\$. *+?\	ab\+ matches ab+
	Matches either one expression or an alternative expression.	.*@example\.(net com) matches against any URI for the domain example.com or the domain example.net

Character	Description	Example
Wildcard and character matching		
.	Matches any single character.	a.c matches aac, abc, azc, a2c, a\$c etc.
\d	Matches a decimal digit character (i.e. 0-9).	a\d matches a1, a2, a3 etc. but not aa, ab etc.
\D	Matches a non-digit character.	a\D matches ab but not a1, a2, a3 etc.
\s	Matches any whitespace character (space, tab, newline).	ab\s matches ab d but not abcd, abxd etc.
\S	Matches any non-whitespace character.	ab\S matches abcd, abxd etc. but not ab d
\w	Shorthand for [a-zA-Z0-9_]. Matches any alphabetical or digit character, or underscore.	\w+ matches bob and bob_jones but not bob.jones. [\w.-]+ matches bob, bob.jones, bob_jones, and bob-jones.
[...]	Matches the characters specified in the brackets. You can specify a range of characters by specifying the first and last characters in the range, separated by a hyphen. You cannot use other special characters within the [] — they will be taken literally.	9[aeiou] matches 9a, 9e, 9i etc. and 9A, 9E, 9I etc, but not 9b, 9c, 9B, 9C etc 9[a-z] matches 9a, 9b, 9z etc. and 9A, 9B, 9Z etc. but not strings such as 91, 99 or 9([0-9#*] matches any single E.164 character (digits 0-9, hash key or asterisk)
[^...]	Matches anything except the set of specified characters.	[^a-z] matches any non-alphabetical character [^0-9#*] matches anything other than an E.164 character
Repetition factors		
*	Matches 0 or more repetitions of the previous character or expression.	ab*c matches ac, abc, abbc, but not ab or abd
+	Matches 1 or more repetitions of the previous character or expression.	ab+c matches abc and abbc, but not ab, ac or abd [a-z.]+ matches any string containing the letters a to z, A to Z, or dots (periods)
?	Matches 0 or 1 repetitions of the previous character or expression.	ab?c matches ac and abc, but not ab, abbc or abd
.*	Matches against any sequence of characters.	a.* matches everything beginning with a
{n}	Matches n repetitions of the previous character or expression.	ab{2}c matches abbc, but not abc or abbbc a\d{3} matches a123 and a789, but not a12 or 456
{n,m}	Matches n to m repetitions of the previous character or expression.	ab{2,4}c matches abbc, abbbc and abbbbc, but not abc or abbbbbc
{n,}	Matches at least n repetitions of the previous character or expression.	ab{2,}c matches abbc, abbbc, abbbbc etc, but not abc
Position matching		
^	Matches the beginning of a line.	^meet\.(.*) matches any string that starts with meet. and places the rest of the string into a group (which could be used in a replace string).
\$	Matches the end of the line.	.*\.com\$ matches any string that ends in .com

Character	Description	Example
(?!...)	Negative lookahead. Defines a subexpression that must not be present immediately after the current match position, for example <code>regex1(?!regex2)</code> where a match is found if <code>regex1</code> matches and <code>regex2</code> does not match.	<p><code>(?!.*@example\.com\$).*</code> matches any string that does not end with <code>@example.com</code></p> <p><code>(?!meet).*</code> matches any string that does not start with <code>meet</code></p> <p><code>meet(?!\.).*</code> matches any string that starts with <code>meet</code> providing it is not followed by a period</p>
(?<!...)	Negative lookbehind. Defines a subexpression that must not be present immediately before the current match position, for example <code>(?<!regex1)regex2</code> where a match is found if <code>regex1</code> does not match and <code>regex2</code> matches.	<code>.*(?<!@)example\.com.*</code> matches any string containing <code>example.com</code> providing it is not preceded by <code>@</code>

Note that this is only a subset of the full range of expressions. For a full description of regular expression syntax see <http://perldoc.perl.org/perlre.html>.

Pattern matching examples

VMR naming patterns

If your videoconferencing rooms have an alias naming convention in the form `oslo1@example.com`, `oslo2@example.com`, `newyork1@example.com` etc, this could be matched with an expression such as:

```
[a-z]+\d@example\.com
```

Or, if the rooms are named 555 followed by exactly three digits, e.g. `555100@example.com` and `555234@example.com`, you could use the expression:

```
555\d{3}@example\.com
```

If your room aliases could follow either pattern, you could merge the two expressions like this:

```
([a-z]+\d|555\d{3})@example\.com
```

Matching an IP address

To match against an IP address, use:

```
^([1-9][0-9]?|1[0-9]{2}|2[01][0-9]|22[0-3])(\.[0-9]|1[0-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])){3}$
```

To match against somebody dialing an IP address (e.g. from an H.323 device or the web app), or someone dialing a SIP URI in the format `IP_address@example.com`, use a **Destination alias regex match** of:

```
^((([1-9][0-9]?|1[0-9]{2}|2[01][0-9]|22[0-3])(\.[0-9]|1[0-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])){3})(@example\.com)?$
```

with a **Destination alias regex replace** string of: `\1`

and this combination of settings will ignore any `@example.com` element and pass just the IP address.

Search and replace examples

Replacing an alias domain

This example transforms any alias that ends in `example.net` into an alias that ends in `example.com`:

Match string: `(.+)(@example\.net$)`

Replace string: `\1@example.com`

This example builds on the previous example by transforming any alias that ends in example.com, example.net or example.co.uk into a common alias that ends in example.com:

Match string: `(.+)(@example\.(com|net|co\.uk)$)`


Replace string: `\1@example.com`

Strip leading 9

This example strips a leading 9 from any all-numeric (0-9) alias (e.g. for integrating with an ITSP phone gateway service):

Match string: `9([0-9]+$)`

Replace string: `\1`

-  If your environment includes a PSTN gateway or uses an ITSP (Internet telephony service provider), consider the potential for toll fraud if you have Call Routing Rules that can route calls to the PSTN gateway or ITSP, or if you allow conference participants to dial out to other participants via the PSTN gateway or ITSP. See [PSTN gateways and toll fraud](#) for more information.

Jinja2 templates and filters

Pexip Infinity uses a subset of the Jinja2 templating language (<https://jinja.palletsprojects.com/en/2.10.x/templates/>) to assist in creating content or deciding on processing logic when configuring VMR Scheduling for Exchange, One-Touch Join, Epic telehealth profiles, writing local policy scripts, and when provisioning VMRs, devices and users from Active Directory via LDAP.

Template content

Jinja2 templates consist of the following elements:

- **literal text** that you want to add to the output or result, such as prefixing every generated VMR name with **meet**.
- **variables** that are either locally defined within the script or template, or that are provided automatically according to the context in which the template is being used, for example there is a `call_info` variable when configuring local policy, or a `givenName` variable when syncing VMRs via LDAP.
- **filters** that can manipulate text or modify the content of variables or text strings, such as `join`, `pex_update` or `pex_to_json`
- **delimiters** such as `{{...}}` and pipes `|` which are used to enclose variables and define filter expressions
- **Jinja statements and control structures** (see <https://jinja.palletsprojects.com/en/2.10.x/templates/#list-of-control-structures>)

Supported Jinja2 filters

Pexip Infinity supports a subset of filters from the Jinja2 templating language. Any Jinja filters that are not listed below have been disabled in Pexip Infinity. See <https://jinja.palletsprojects.com/en/2.10.x/templates/#list-of-builtin-filters> for more information about these filters.

<code>abs</code>	<code>float</code>	<code>last</code>	<code>replace</code>	<code>truncate</code>
<code>capitalize</code>	<code>format</code>	<code>length</code>	<code>round</code>	<code>upper</code>
<code>default</code>	<code>int</code>	<code>lower</code>	<code>striptags</code>	
<code>first</code>	<code>join</code>	<code>range</code>	<code>trim</code>	

To use a filter you would typically follow the syntax `{{<source_value>|<filter_name>}}`.

In most cases the `<source_value>` is likely to be a variable, for example `{{givenName|upper}}`, although it could be one or more literal values, for example `{{ [1, 2, 3, 4]|join }}`.

Some filters take parameters, for example `{{sn|truncate(5)}}`. You can also use multiple filters in the same expression, for example `{{sn|truncate(5)|upper}}`.

The `trim` filter is often used. This trims leading and trailing whitespace from the string held in the `<source_value>`.

Example usage: `{{title|trim}}`

If the `title` field contained " Project Manager ", this would be converted to "Project Manager".

The `replace` filter is also often used. This replaces one string with another string. The first argument of the filter is the substring that should be replaced, the second is the replacement string. If the optional third argument count is given, only the first count occurrences are replaced.

Example usage: `{{ department|replace("Personnel", "HR") }}`

If the `department` field contained "Personnel Department Personnel", this would be converted to "HR Department HR".


Example usage: `{{ department|replace("Personnel", "HR", 1) }}`

In this case a count of 1 is specified, thus if the `department` field contained "Personnel Department Personnel", it would be converted to "HR Department Personnel".

For more complicated search and replace patterns, use the custom Pexip `pex_regex_replace` filter described below.

Custom Pexip filters

In addition to the Jinja filters, Pexip also provides the following custom filters, which are typically used to manipulate data:

Filter	Description and example usage
<code>pex_base64</code>	Performs Base64 encoding on the input field.
<code>pex_clean_phone_number</code>	This extracts only +0123456789 characters (and removes ()&%#@ '";, A-Z,a-z etc).
<code>pex_debug_log (message)</code>	<p>The <code>pex_debug_log</code> filter can be used to help debug your script. It writes debug messages to the Pexip Infinity support log. You can include literal text and variables.</p> <p> To avoid filling the support log and causing it to rotate, remove all <code>pex_debug_log</code> filters from your scripts as soon as they are working correctly.</p>
<code>pex_find_first_match(string_list, 'find_regex')</code>	This extracts from the list the first value that matches the specified regex.
<code>pex_hash</code>	Performs a hash of a field.
<code>pex_head (maxlength)</code>	Returns, at most, the first maxlength characters from the input field.
<code>pex_in_subnet</code>	Tests whether a given address is within one or more subnets. It takes as input the address you want to test, and one or more subnet ranges, and returns either True or False.
<code>pex_md5</code>	Applies an MD5 hash to the input field.
<code>pex_now(timezone)</code>	<p>The <code>pex_now</code> filter takes an optional parameter of a timezone description e.g. 'UTC', 'Asia/Tokyo', or 'US/Eastern' and returns the current date and time for that timezone. UTC is assumed if a timezone is not provided.</p> <p>The resulting available attributes are <code>year</code>, <code>month</code>, <code>day</code>, <code>hour</code>, <code>minute</code>, <code>second</code> and <code>microsecond</code>.</p> <p>Example usage:</p> <pre>{% set now = pex_now("Europe/London") %} {% if now.month == 2 and now.day == 29 %}</pre>
<code>pex_random_pin (length)</code>	Generates a random PIN of the given length. Note that this filter does not take any input.
<code>pex_regex_replace ('find_regex', 'replace_string')</code>	<p>This performs a regex find and replace.</p> <p>Example usage: <code>{{ mail pex_regex_replace('@.+','@otherdomain.com') }}</code></p> <p>This example takes as input an email address contained in the <code>mail</code> variable and changes the domain portion of the address to <code>@otherdomain.com</code>. For example, it will transform <code>user1@domainA.com</code> to <code>user1@otherdomain.com</code>, and <code>user2@domainB.com</code> to <code>user2@otherdomain.com</code> etc.</p>
<code>pex_regex_search ('regex pattern', 'string_to_search')</code>	<p>This performs a regex search for the first location that matches the pattern and returns the regex groups.</p> <p>Example usage:</p> <pre>{% set groups = pex_regex_search("[a-z0-9-]+@[a-z0-9-]+.com)", "example string with someone@example.com") %} {% if groups %} {{ groups[0] }}@{{ groups[1] }} % endif %</pre> <p>This example takes as input a string containing an email address and extracts the email using two regex groups.</p>
<code>pex_require_min_length(length)</code>	<p>This validates that the input string field has the specified minimum length.</p> <p>Syntax: <code>{{ some_string pex_require_min_length(2) }}</code></p>

Filter	Description and example usage
pex_reverse	This reverses the characters in the input field.
pex_strlen	Returns the length of string. The basic usage syntax is: <pre>{% set some_length = "Example" pex_strlen %} {# sets a variable named some_length to the value 7 #}</pre>
pex_tail(maxlength)	Returns, at most, the last maxlength characters from the input field.
pex_to_json	Converts a Python dictionary variable into JSON format.
pex_to_uuid	Converts a base64 string to a UUID.
pex_update	Updates Python dictionary variables.
pex_url_encode	This filter creates URL parameters that are safely URL-encoded.
pex_uuid4()	This generates a uuid (universally unique identifier). Note that this filter does not take any input.

Encryption methodologies

Pexip nodes

All communication links between the Management Node and Conferencing Nodes, and between Conferencing Nodes, use an IPsec transport with the following settings:

- 256-bit AES-GCM for encryption
- a 4096 bit Diffie-Hellman modulus for key exchange.

No other ciphers, hashes or moduli are permitted.

These settings apply to both the initial channel set up for key exchange ([ISAKMP](#)) and the secondary channel over which application data is transported ([ESP](#)).

Inter-node traffic is restricted to only protocols that are expected for the successful operation of Pexip Infinity, including but not necessarily limited to call signaling, media, status, and configuration information; any unexpected traffic/protocols are dropped.

Endpoints

Encrypted connections between Pexip Infinity and endpoints use:

- AES 128-bit or 256-bit encryption for media
- TLS for SIP call control (for more information, see [Managing TLS and trusted CA certificates](#))
- SRTP for SIP media
- H.235 for H.323 media

Infinity Connect (web/desktop/mobile) clients use:

- HTTPS TLS for signaling
- DTLS-SRTP for WebRTC media

Interoperability

We endeavor to make Pexip Infinity interoperable with all relevant standards-based devices. For the most up-to-date information on the devices and software versions with which Pexip Infinity is known to work, along with any known issues, see

<https://docs.pexip.com/admin/interoperability.htm>.

We welcome your feedback on any known issues with any devices, and we would also like to hear of any other devices that you have used with Pexip Infinity. Please send your comments to your Pexip authorized support representative.

Supported RFCs

Pexip Infinity supports the following RFCs:

- RFC 1889 RTP: A Transport Protocol for Real-time Applications
- RFC 2190 RTP Payload Format for H.263 Video Streams
- RFC 2429 RTP Payload Format for the 1998 Version of ITU-T Rec. H.263 Video (H.263+)
- RFC 2782 DNS RR for specifying the location of services (DNS SRV)
- RFC 2790 Host Resources MIB
- RFC 2976 The SIP INFO Method
- RFC 3261 SIP: Session Initiation Protocol
- RFC 3263 Locating SIP Servers
- RFC 3264 An Offer/Answer Model with SDP
- RFC 3550 RTP: A Transport Protocol for Real-Time Applications
- RFC 3581 Symmetric Response Routing
- RFC 3605 RTCP attribute in SDP
- RFC 3711 The Secure Real-time Transport Protocol (SRTP)
- RFC 3840 Indicating User Agent Capabilities in SIP
- RFC 3890 A Transport Independent Bandwidth Modifier for SDP
- RFC 3891 SIP "Replaces" Header
- RFC 3984 RTP Payload Format for H.264 Video
- RFC 4320 Actions Addressing Identified Issues with the Session Initiation Protocol's (SIP) Non-INVITE Transaction
- RFC 4321 Problems Identified Associated with the Session Initiation Protocol's (SIP) Non-INVITE Transaction
- RFC 4566 SDP: Session Description Protocol
- RFC 4568 SDP: Security Descriptions for Media Streams
- RFC 4574 The Session Description Protocol (SDP) Label Attribute
- RFC 4582 The Binary Floor Control Protocol
- RFC 4583 SDP Format for BFCP Streams
- RFC 4585 Extended RTP Profile for RTCP-Based Feedback
- RFC 4587 RTP Payload Format for H.261 Video Streams
- RFC 4629 RTP Payload Format for ITU-T Rec. H.263 Video
- RFC 4733 RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals
- RFC 4796 The SDP Content Attribute
- RFC 5168 XML Schema for Media Control
- RFC 5245 Interactive Connectivity Establishment (ICE)
- RFC 5389 Session Traversal Utilities for NAT (STUN)
- RFC 5577 RTP Payload Format for ITU-T Recommendation G.722.1
- RFC 5763 Framework for Establishing a Secure Real-time Transport Protocol (SRTP) Security Context Using Datagram Transport Layer Security (DTLS)
- RFC 5766 Traversal Using Relays around NAT (TURN)
- RFC 6026 Correct Transaction Handling for 2xx Responses to Session Initiation Protocol (SIP) INVITE Requests
- RFC 6125 Representation and Verification of Domain-Based Application Service Identity
- RFC 6416 RTP Payload Format for MPEG-4 Audio/Visual Streams
- RFC 7714 AES-GCM Authenticated Encryption in the Secure Real-time Transport Protocol (SRTP)
- draft-ietf-bfcpbis-rfc4582bis-10.txt
- draft-ietf-bfcpbis-rfc4583bis-08.txt
- draft-ietf-avt-rtp-h264-params-01.txt
- draft-ietf-payload-rtp-opus-01.txt

- [draft-ietf-payload-rtp-vp8-10.txt](#)
- [draft-ietf-mmusic-sdp-g723-g729-04.txt](#)

Patents

This table is intended to serve as notice under 35 U.S.C. § 287(a).

Product	Patent
Pexip Infinity	US 8971407
Pexip Infinity	US 8976225
Pexip Infinity	US 9407933
Pexip Infinity	US 9118808
Pexip Infinity	US 10382337
Pexip Infinity	US 10873745

Accessibility

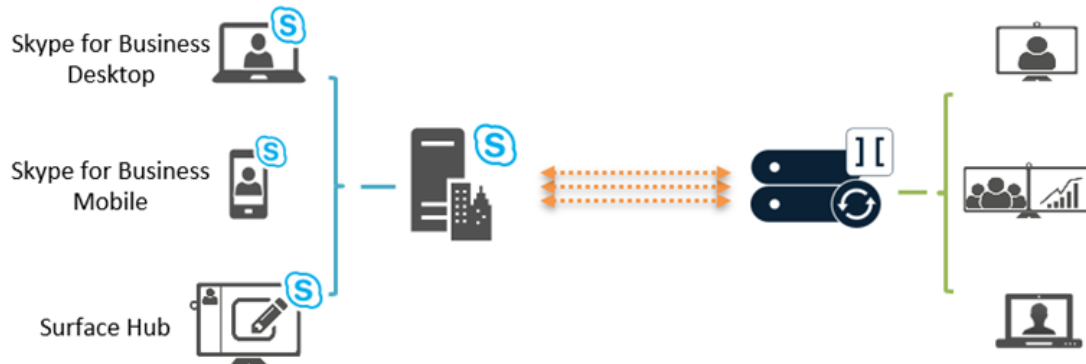
The following features are designed to improve accessibility of the Pexip Infinity user experience, including the Infinity Connect clients, in support of AAA WCAG 2.0 compliance.

Feature	More information
High contrast mode	The High Contrast option increases contrast between the foreground and background elements of the user interface, making them more legible.
Progress animations and customizable spinners	Colors and contrast can be customized through branding portal.
Resizing Text	Text can be resized to 200% without loss of content or function.
Timeline	The timeline displays a graphical overview of the meeting context and events.
Events stream	All significant events and chats are displayed textually in the events panel.
Screen reader support	Screen readers can access the Events stream live.
Navigation and focus	Navigation has been designed to streamline keyboard input.
Help	Tooltips are used throughout the client to create context and provide help.
Audio prompts	Audio prompts complement splash screen messaging. These are customizable.
Text overlay	Text overlays displaying participant names can be enabled.
Alarms	The presence of active alarms is indicated by a flashing blue icon on the Management Node Administrator interface.
Choice of colors	Colors used in the Administrator interface have been chosen to provide maximum accessibility and contrast. For example, in the Administrator Logs and Support logs, warnings and errors are highlighted with an orange or blue background respectively; Live View search is yellow and blue.

Using Microsoft Skype for Business / Lync with Pexip Infinity

Pexip Infinity allows Microsoft Skype for Business and Lync* users to meet with other people regardless of the system they are using – Skype for Business / Lync, web browsers or traditional video conferencing systems. All participants can enjoy wideband audio, high definition video and cross-platform presentation sharing.

It can be integrated with SfB/Lync as part of an existing, on-premises SfB/Lync environment inside an enterprise network, or as a standalone Pexip environment deployed in a public DMZ that enables direct federation with remote SfB/Lync environments, or as a hybrid deployment where SfB/Lync users may be homed either on-premises or in Office 365.



Pexip Infinity enables full interoperability between Microsoft's H.264 SVC/RTV/RDP and H.263, H.264, VP8 (WebRTC) and BFCP/H.239 for truly seamless video and content sharing in any-to-any configurations, such as multiparty conferences.

In addition to enabling SfB/Lync participants to join conferences hosted on Pexip Infinity, Pexip Infinity can act as a gateway between SfB/Lync and standards-based endpoints. This enables SfB/Lync clients to receive and initiate point-to-point calls with H.323/SIP endpoints and registered Infinity Connect clients, and invite those devices into a SfB/Lync meeting while retaining the native meeting experience on each device.

* Note that where this documentation refers to "SfB/Lync", it represents both Microsoft Skype for Business and Lync unless stated otherwise.

Architecture options

Pexip Infinity can be integrated with Microsoft Skype for Business and Lync in three different ways:

- As part of an existing, on-premises Lync environment inside an enterprise network, referred to as **on-premises deployment**.
To integrate Pexip Infinity with an existing, on-premises SfB/Lync environment, one or more SIP domains are statically routed from the SfB/Lync environment towards one or more Pexip Infinity Conferencing Nodes. Then, when a SfB/Lync user dials a conference alias, such as meet.john@vc.example.com, or the alias of a standards-based endpoint, the user is placed into the appropriate Pexip-hosted conference. The SfB/Lync user can also pin one or more such aliases to their contact list for easy access later.
- As a standalone Pexip environment deployed in a public DMZ, referred to as **public DMZ deployment**.
As Pexip Infinity supports SfB/Lync natively, it can be deployed to enable SfB/Lync interoperability without having any existing, on-premises SfB/Lync infrastructure. In such a deployment, Pexip Infinity can federate directly with remote SfB/Lync environments (on-premises environments as well as SfB/Lync Online/Office 365), without the need for a local SfB/Lync environment.
- As a **hybrid deployment** which is a mix of on-premises and Office 365 deployments where users may be homed in either environment. A hybrid deployment has the same configuration requirements as a **public DMZ deployment**.

You will typically choose one of these methods, depending on requirements and preference.

Pexip Infinity as a SfB/Lync gateway

Pexip Infinity can act as a gateway between SfB/Lync and standards-based endpoints. This enables SfB/Lync clients to:

- invite H.323/SIP endpoints and registered Infinity Connect clients into a Sfb/Lync meeting via manual dialout or drag and drop from the contacts list
- use the Infinity Gateway to route incoming calls directly into an ad hoc or scheduled Sfb/Lync meeting
- when dialed into a Pexip VMR conference, invite other Sfb/Lync or external contacts into that same Pexip VMR (this creates a new Sfb/Lync meeting which is merged with the existing Pexip VMR)
- receive and initiate person-to-person calls with standards-based devices, and then optionally add other participants (to escalate to a multipoint Sfb meeting).

Skype for Business / Lync**More information**

For full information on using Microsoft Skype for Business / Lync with Pexip Infinity, see [Integrating Microsoft Skype for Business / Lync with Pexip Infinity](#).

When is a reverse proxy, TURN server or STUN server required?

i Since version 16 of Pexip Infinity, we recommend that you deploy Proxying Edge Nodes instead of a reverse proxy and TURN server if you want to allow externally-located clients to communicate with internally-located Conferencing Nodes.

If you do not want to deploy Proxying Edge Nodes, and all of your Conferencing Nodes are privately addressed, you will need to use a reverse proxy and a TURN server to allow external endpoints such as Infinity Connect clients to access your Pexip Infinity services, and you may need to use a TURN server for Skype for Business / Lync clients. A TURN server can also act as a STUN server, however, in some Pexip Infinity deployment scenarios where the TURN server is deployed inside your enterprise firewall, you may need to configure a separate, external STUN server.

When connecting to a privately-addressed Conferencing Node, Infinity Connect WebRTC clients that are behind a NAT may also use a STUN server to find out their public NAT address.

The following table shows when a reverse proxy, TURN server or STUN server needs to be deployed (if you are not using Proxying Edge Nodes). When used, they must be publicly accessible, and routable from your on-premises Conferencing Nodes.

External endpoint / client	Conferencing Node addresses	Reverse proxy	TURN server	STUN server (for Conferencing Nodes)	STUN server (for WebRTC clients behind NAT)
Infinity Connect WebRTC clients	Private (on-premises)	✓	✓	✓ (if the TURN server is inside the firewall)	✓
Skype for Business / Lync clients *	Private (on-premises)	-	✓ (only required if internal Conferencing Node cannot route to the public-facing interface of the SfB/Lync Edge server)	✓ (if the TURN server is inside the firewall)	✓
Any endpoint / client	Publicly reachable — either directly or via static NAT	-	-	-	-

* Also requires a Skype for Business / Lync Edge Server when Conferencing Nodes are privately addressed.

Note that you may still want to deploy a reverse proxy in front of your Proxying Edge Nodes if, for example, you want to:

- host customized Infinity Connect web app content
- use it as a load balancer for Pexip's VMR Scheduling for Exchange service, to proxy requests from Outlook clients to Conferencing Nodes.

Integrating with streaming and recording services

This section provides step-by-step instructions on how to integrate with popular streaming services. For an overview of streaming with Pexip Infinity, see [Streaming and recording a conference](#).

Streaming a conference to YouTube

This guide explains how to stream a conference being held in a Pexip Virtual Meeting Room or Virtual Auditorium to YouTube.

YouTube prerequisites

Before you can obtain an RTMP streaming URL from YouTube, you must ensure that you have a verified YouTube account, and that the account is enabled for live events.

1. From your YouTube account settings page, select **Channel status and features** (www.youtube.com/features).
2. If your **Account status** is not verified, select **Verify** and follow the YouTube instructions.
3. If **Live streaming** is not enabled, click **Enable**.

Note that enabling your first live stream may take up to 24 hours.

Setting up streaming

To stream a conference to YouTube, you must first obtain a live streaming URL via YouTube. You then initiate a call from the VMR to the YouTube URL, by adding the YouTube URL as a conference participant.

There are two ways in which you can obtain an RTMP streaming URL from YouTube:

- Use Pexip's own utility at <https://yt.pexip.com>. This method simplifies the generation process and automatically uses the appropriate settings. However, it is only suitable if you want to set up a **single stream** with a single-use stream key.
- Generate your URL directly from within your YouTube account at www.youtube.com/my_live_events. You must use this method if you want to use a reusable stream key or set up **dual streams** within the same broadcast (i.e. a "second camera" in YouTube terms).

When generating a URL directly from within YouTube, the **Privacy** setting is **Public** by default, so we recommend that you change this to **Unlisted**.

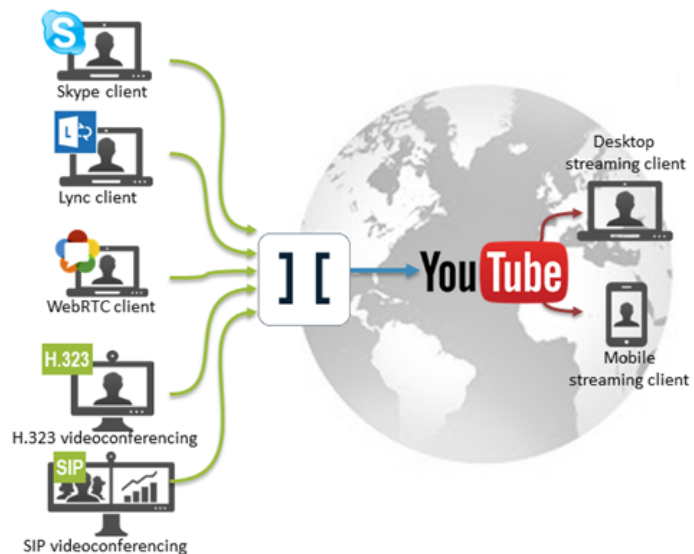
Note that the live stream will have a 20-30 second delay. This is because YouTube buffers the stream so that it can tolerate brief connection losses and to ensure a good consistent experience. This is standard streaming behavior.

The appropriate procedures for obtaining your streaming URL, adding single or dual-streamed participants to your conference, and streaming at Full HD (1080p) are described below.

Setting up your URL via Pexip's utility and adding a single streaming participant

This procedure explains how to use Pexip's own utility to request a URL for a single YouTube RTMP stream on your behalf, and how to add that stream as a participant to your Pexip conference.

Note that this utility requests a single-use stream key. If you want to reuse a previous stream key/URL (for example if you want to use this URL as an Automatically Dialed Participant) then follow the instructions for setting up your URL [from within YouTube](#) (but do not set up a second camera).

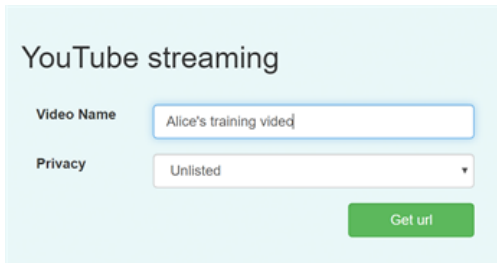


Obtaining the YouTube streaming URL

1. Go to <https://yt.pexip.com>.
2. Enter a **Video Name** — this is the name that will appear in YouTube.
3. Select a **Privacy** level:
 - **Unlisted**: viewers must know the streaming URL to see the stream.
 - **Public**: anybody can find the stream on YouTube. This is not recommended unless you are streaming very public content.
 - **Private**: restricts access to only people that you have explicitly allowed to view the stream.

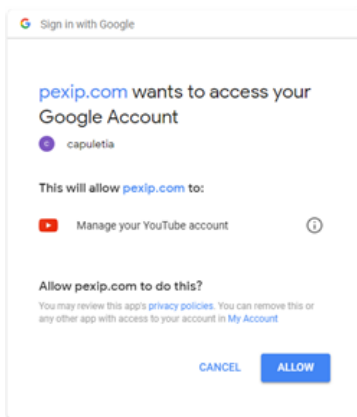
Default: **Unlisted**.

4. Select **Get url**.



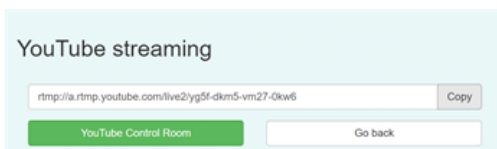
The screenshot shows a web form titled "YouTube streaming". It has two input fields: "Video Name" with the text "Alice's training vided" and "Privacy" with a dropdown menu set to "Unlisted". Below these fields is a green button labeled "Get url".

5. If you are not already signed in to a Google Account, you must either sign in or select an account.
6. At the prompt, Allow pexip.com to access your Google account:



The screenshot shows a Google sign-in prompt. It says "Sign in with Google" and "pexip.com wants to access your Google Account". Below this, it lists the permissions: "This will allow pexip.com to: Manage your YouTube account". At the bottom, there are "CANCEL" and "ALLOW" buttons.

7. The streaming URL will be generated and displayed.



The screenshot shows the "YouTube streaming" form again, but now it displays a long RTMP URL in the "Video Name" field: "rtmp://a.rtmp.youtube.com/live2/yg5f-dkm5-vm27-dkw6". There is a "Copy" button next to the URL. Below the URL field are two buttons: "YouTube Control Room" and "Go back".

If you receive a "The user is not enabled for live streaming" error message, this means that you either do not have a verified YouTube account, or that the account is not enabled for live events.

8. Copy the rtmp:// address. Leave this browser window open.

Adding the participant URL and enabling streaming

Now that you have the YouTube streaming URL, you can initiate a call from the Virtual Meeting Room to the YouTube URL, and then begin streaming.

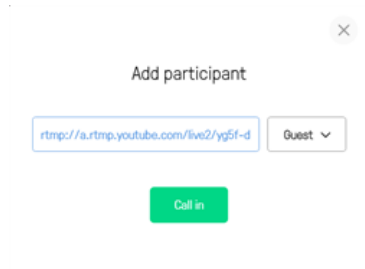
1. Initiate a call from the Virtual Meeting Room to the streaming address. This is done by adding the streaming address as a conference participant. You can do this either from the [Pexip Infinity Administrator interface](#) or from an [Infinity Connect client](#) connected to the VMR.



When using the Administrator interface, use the following settings:

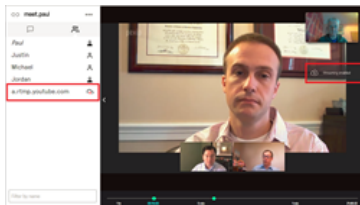
- **Protocol:** *RTMP*
- **Address:** the YouTube streaming URL.
- **Role:** we recommend selecting *Guest* (so that the streaming participant is not shown to other Guests in a Virtual Auditorium layout, and so that it does not keep a conference alive when all other Hosts have left).
- **Streaming:** select this option.

When using an Infinity Connect client, use the following settings:

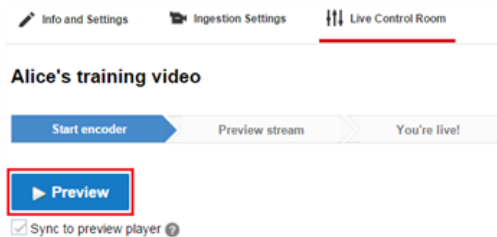
- **Participant details:** enter your YouTube streaming URL e.g. `rtmp://a.rtmp.youtube.com/live2/yg5f-dkm5-vm27-0kw6`
RTMP authentication is supported; in this case credentials are included in the URI using the syntax `rtmps://username:password@host/....`
Note that a [suitable Call Routing Rule](#) is required when dialing out to a streaming service via Infinity Connect clients.
- **Role:** we recommend selecting *Guest*.



When Pexip Infinity has placed the call to the streaming service, the **Streaming enabled** icon  is displayed, and for Infinity Connect users the streaming participant appears in the participant list with a streaming badge  next to its name:



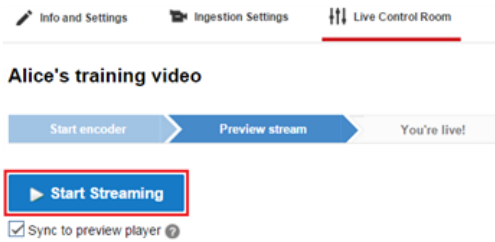
- Wait for a few seconds, then from within your YouTube account, go to your **Live Control Room**. (You can select the green link below the `rtmp://` address that you copied when using Pexip's URL generator.)
- Select **Preview** and confirm.



After a few seconds you will be able to **Play** the Preview stream in the video window below.

Note that this is your preview only — at this stage the stream is not being broadcast. The stream has a 20-30 second delay.

- Click **Start streaming** and confirm, to start broadcasting.



5. You are now streaming to anyone who is allowed to access or find your streams (according to your **Privacy** settings). You can optionally **Play the Public View** of the stream (in the window below the Preview stream).
6. You can select **View on Watch Page** (top right of your **Live Control Room** page) to see the normal YouTube view. This is how it appears to users who are watching the live stream, and is the URL that you should share.

Setting up your URL from within YouTube and adding a dual streaming participant

This procedure explains how to use YouTube to generate a streaming URL. It also shows how to set up dual streams, and how to add those streams as a participant to your Pexip conference. Using YouTube to generate the streaming URL also allows you to reuse a previous stream key/URL (for example if you want the URL to be automatically dialed whenever a particular VMR is used).

Note that:

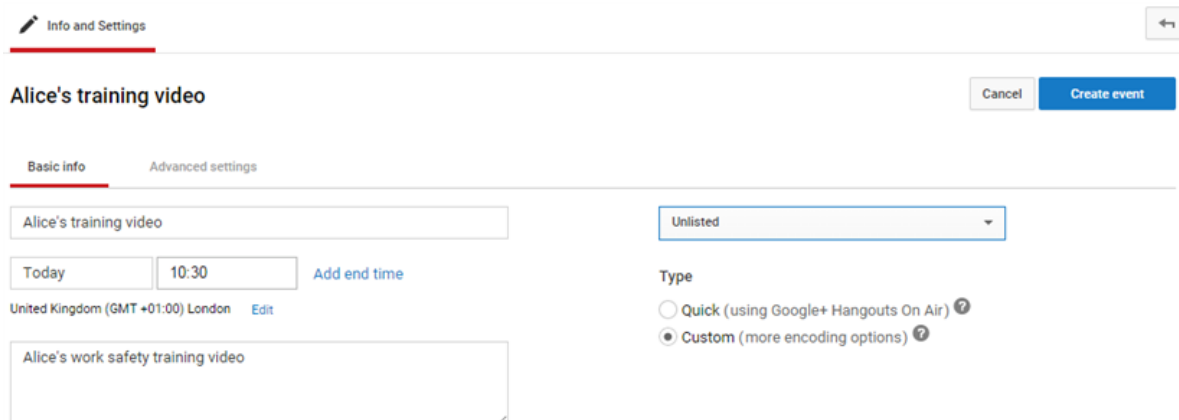
- The YouTube **Privacy** setting is **Public** by default, so we recommend that you change this to **Unlisted**.
- You cannot use the Infinity Connect clients to add dual-streaming participants — you must use the Administrator interface.

Obtaining the YouTube streaming URL

To generate your streaming URL directly from within YouTube:

1. In your YouTube account go to **Live Streaming > Events** (www.youtube.com/my_live_events).
2. Select **New live event** (at the top right of the page).
3. Enter the **Basic info**:

Title	The title of the video.
Start time	Set the start time to 30 minutes in the past (so that it is available instantly).
Privacy	Select a privacy level. We recommend Unlisted , which means that viewers must know the streaming URL to see the stream.
Type	Select Custom .



4. Select **Create event**. You are taken to the **Main Camera** tab.
5. Configure the main video stream:

- a. Optionally, you can upload a Thumbnail for this stream.
- b. Select the type of stream key. Your options are:
 - **Single-use stream key**: typically used for one-off events.
 - **Reusable stream key**: typically used for recurring events or when you want to use this streaming URL as an Automatically Dialed Participant. If you select **Reusable stream key** you can either create a new stream key, or select a stream key you have previously created.
- c. From the Select your encoder drop-down list, select **Other encoders**.


The screenshot shows the 'Alice's training video' configuration page in Pexip Infinity. The page has a top navigation bar with 'Info and Settings', 'Ingestion Settings', 'Cards', and 'Live Control Room'. On the right, there are buttons for 'View on Watch Page' and a back arrow. Below the navigation bar, there are 'Cancel' and 'Save changes' buttons. The main content area is titled 'Alice's training video' and has a 'Main Camera' tab selected, with an 'Add a Camera' link next to it. Under the 'Main Camera' tab, there is a 'Thumbnail' section with a placeholder image and a 'Browse' button. Below this is the 'Select type of stream key' section, which has two radio buttons: 'Single-use stream key' (selected) and 'Reusable stream key' (with a help icon). Below that is the 'Select your encoder' section, which has a dropdown menu set to 'Other encoders'. Below the dropdown, there are four numbered steps: 1. Configure your encoder (with a link to 'Recommended bitrates and settings'), 2. Copy and paste into your encoder (with instructions to enter stream names and URLs), 3. Start your encoder (with instructions to start sending the video stream), and 4. Go to the Live Control Room (with instructions to preview and start the event). Under step 2, there are three input fields: 'Stream Name' (containing '1020-q5ah-63d9-dwc9'), 'Primary Server URL' (containing 'rtmp://a.rtmp.youtube.com/live2'), and 'Backup Server URL' (containing 'rtmp://b.rtmp.youtube.com/live2?backup=1').

6. Configure the second stream (if you want dual streams):
 - a. Select **Add a Camera** (next to the Main Camera tab).
 - b. Optionally, you can upload a **Thumbnail** for this stream.
 - c. Enter the Camera Name e.g. "Training presentation".
 - d. Select the type of stream key, either **Single-use** or **Reusable**.
If you select **Reusable**, choose a different stream from the one you are using for the Main Camera.
 - e. From the **Select your encoder** drop-down list, select **Other encoders**.

Main Camera **Camera 2** Add a Camera

Thumbnail

Please upload as large an image as possible (suggested: 1280 x 720) since the image will also be used as the preview image when your event is embedded on other sites. You can upload a JPG, GIF, BMP or PNG file. Maximum file size is 2 MB.




Camera Name*

Select type of stream key

Choose between a single-use or reusable stream key. Reusable keys are named and allow for easier stream setup next time, recurring events or simultaneous events of the same quality.

☒ Single-use stream key

☐ Reusable stream key 

Select your encoder

YouTube Live provides support for a variety of encoders. Select one of the encoder options below and follow the instructions.

- 1. Configure your encoder**
[Recommended bitrates and settings](#)
- 2. Copy and paste into your encoder**
Enter the stream names and URLs in the configuration options of your encoding software.

Stream Name

Primary Server URL

Backup Server URL
- 3. Start your encoder**
In your encoder, start sending us your video stream.
- 4. Go to the Live Control Room.**
You can preview and start your event from the [Live Control Room](#).

7. Select **Save changes**.
8. Produce the RTMP URL for your primary video stream as **Primary Server URL/Stream Name** of the Main Camera:
 - a. Go to the **Main Camera** tab.
 - b. Take the **Primary Server URL**, for example `rtmp://a.rtmp.youtube.com/live2`.
 - c. Append a / (slash).
 - d. Then append the **Stream Name**, for example `1020-q5ah-63d9-dwc9`.

In this example, the RTMP URL is `rtmp://a.rtmp.youtube.com/live2/1020-q5ah-63d9-dwc9`.

You will use this RTMP URL as the first address of the new participant in your Pexip conference.
9. Produce the RTMP URL for your presentation stream as **Primary Server URL/Stream Name** of Camera 2:
 - a. Go to the **Camera 2** tab.
 - b. Take the **Primary Server URL**, for example `rtmp://a.rtmp.youtube.com/live2`.
 - c. Append a / (slash).
 - d. Then append the **Stream Name**, for example `9qem-q51r-mrp9-0qdz`.

In this example, the RTMP URL is `rtmp://a.rtmp.youtube.com/live2/9qem-q51r-mrp9-0qdz`.

You will use this RTMP URL as the second stream address of the new participant in your Pexip conference.



Adding dual participant URLs and enabling streaming

After you have set up dual YouTube streaming URLs, you can initiate a call from the VMR to the YouTube URLs, and then begin streaming:

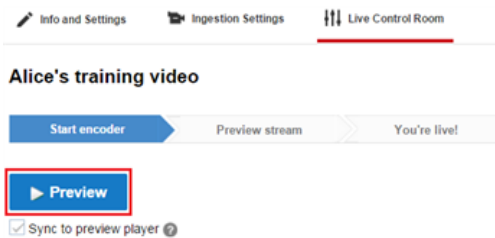
1. Add the YouTube streaming URLs as a new RTMP participant in the VMR you want to stream. For dual streams, you can only do this from the [Pexip Infinity Administrator interface](#):
 - a. Go to **Status > Conferences** and select the name of the Virtual Meeting Room or Virtual Auditorium being used.
 - b. At the bottom left of the screen, select **Dial out to participant**.
 - c. Complete the following fields (other fields can be left to their default values):

Field	Description
System location	Select the system location from which the call will be placed.
Participant alias	Enter the streaming URL of the primary video stream (from the Main Camera tab in the YouTube Ingestion settings) as the address to dial.
Route this call	Select <i>Manually</i> .
Protocol	Select <i>RTMP</i> .
Role	We recommend selecting <i>Guest</i> .
Streaming	Select this check box.
Dual stream (presentation) URL	Enter the second presentation streaming URL (from the Camera 2 tab) into the address field that appears.

- d. Select **OK**.


When Pexip Infinity has placed the call to the streaming service, the **Streaming enabled** icon  is displayed, and for Infinity Connect users the streaming participant appears in the participant list with a streaming badge  next to its name:

2. Wait for a few seconds, then from within your YouTube account, go to your **Live Control Room**.
3. Select **Preview** and confirm.

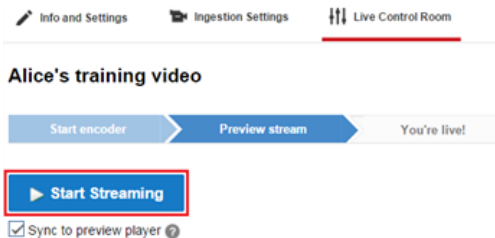


After a few seconds you will be able to **Play** the Preview stream (of the primary video stream).

Note that this is your preview only — at this stage the stream is not being broadcast. The stream has a 20-30 second delay.

-  The second stream (Camera 2) may show "noise" and report video format issues. However, this stream should display correctly when you start live streaming.

4. Click **Start streaming** and confirm, to start broadcasting.



You do not have to preview/start the "Camera 2" presentation stream.

5. You are now streaming to anyone who is allowed to access or find your streams (according to your **Privacy** settings). You can optionally **Play the Public View** of the stream (in the window below the Preview stream).
6. You can select **View on Watch Page** (top right of your **Live Control Room** page) to see the normal YouTube view. This is how it appears to users who are watching the live stream, and is the URL that you should share.

When you have dual streams, YouTube viewers can switch views between the main camera and the second presentation stream by selecting the **Switch Camera** icon at the bottom right of the main window:




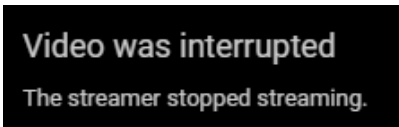
If nobody is currently presenting, Pexip Infinity sends a placeholder image on the presentation stream.

Stopping streaming

This procedure explains how to stop streaming your conference (for either single-streamed or dual-streamed conferences).

To stop streaming your Pexip conference:

1. Disconnect the streaming participant from the Virtual Meeting Room.
 - When using Infinity Connect: the streaming participant appears in the participant list with a streaming badge  and a name that is typically in the format `a.rtmp.youtube.com`.
 - When using the Administrator interface (**Status > Conference**, and then select the conference): the participant alias is the streaming URL with an alias typically in the format `rtmp://a.rtmp.youtube.com.<etc.>`.
2. The Live Control Room will report that the stream status has **No Data**, and the YouTube public stream will now display:



Note that while the stream is still open in your YouTube **Live Control Room**, you can restart streaming by adding the RTMP URL to the conference again as a new participant.

3. In your YouTube **Live Control Room**, select **Stop Streaming** and confirm.



4. A few minutes after your stream has ended, it will appear under **Video Manager > Videos**.
From here you can delete the video if you do not want it to be available for later use, or you can change its privacy settings. You can also use the YouTube video editor to combine multiple recordings, or remove parts of a recording before you publish it etc. (If the video is slow to appear in the **Videos** list, you can also check its content by going to **Video Manager > Live Events** and viewing all **Completed** events.)

Note that if you have produced two recordings (because you selected **Dual Stream**), you must manage each recording separately. When playing back the recordings, YouTube does not provide any mechanism to synchronize them to each other.

Adding a presentation stream when single streaming is already in progress

This procedure explains how to change an existing single streamed RTMP participant into a dual streamed participant.

YouTube ingestion settings cannot be changed after streaming has started. Therefore the only way to add a separate presentation stream after streaming has already started is set up a new, second stream (rather than a second camera on the existing stream) for the presentation channel.

Unlike setting up dual streams from the outset, this time the two streams are completely separate YouTube events and must be viewed in separate browser windows/tabs.

If you have already added a single stream participant to an in-progress conference but want to change this to a dual stream:

1. Go to <https://yt.pexip.com> and obtain a second streaming URL. Enter a **Video Name** (e.g. "Alice's training video presentation content") and select **GET url**.

2. From the Administrator interface, go to **Status > Conferences** and select the name of the Virtual Meeting Room or Virtual Auditorium being used.
3. Ensure that you have a copy of the existing streaming URL for the video channel:
 - The streaming participant appears in the participant list with a name that is typically in the format **a.rtmp.youtube.com**.
 - You can select the participant from the participant list and copy the rtmp URL/alias.
4. Disconnect the existing streaming participant from the Virtual Meeting Room.

Note: do **not** stop streaming in the YouTube **Live Control Room**.

5. Add the streaming participant back in to the conference again, but this time include the second streaming URL.
At the bottom left of the screen of the conference status screen, select **Dial out to participant** and complete the following fields (other fields can be left to their default values):

Field	Description
System location	Select the system location from which the call will be placed.
Participant alias	Enter the original, existing streaming URL (from step 3).
Route this call	Select Manually .
Protocol	Select RTMP .
Role	We recommend selecting Guest .
Streaming	Select this check box.
Dual stream (presentation) URL	Enter the second streaming URL (from step 1).

Select **OK** to add the two streams back into the conference.

6. Start broadcasting the second stream in YouTube:
 - a. Go to the **Live Control Room** for the second stream.
(You can select the green link below the rtmp:// address on the Pexip URL generator screen from step 1.)
 - b. Select **Preview** and confirm.
 - c. **Play** the Preview stream.
 - d. Click **Start streaming** and confirm, to start broadcasting the presentation stream.
7. You are now streaming the presentation channel in addition to the original video channel to anyone who is allowed to access or find your streams (according to your Privacy settings). You can optionally **Play** the **Public View** of the presentation stream.
8. You can select **View on Watch Page** (top right of your **Live Control Room** page) to see the normal YouTube view and to obtain the URL of the presentation stream that you should share with the conference viewers, alongside the existing video stream URL.

Note that the original conference stream will experience a break in content for the period of time between disconnecting the existing streaming participant from the VMR and adding it back in again as one of the dual streamed participants. The YouTube **Live Control Room** for the original stream will report that no data is being received, but this will be resolved automatically when the streaming participant is added back into the conference (providing the original streaming URL is used).

Streaming at Full HD (1080p)

If you want to stream at Full HD (1080p) resolution you must:

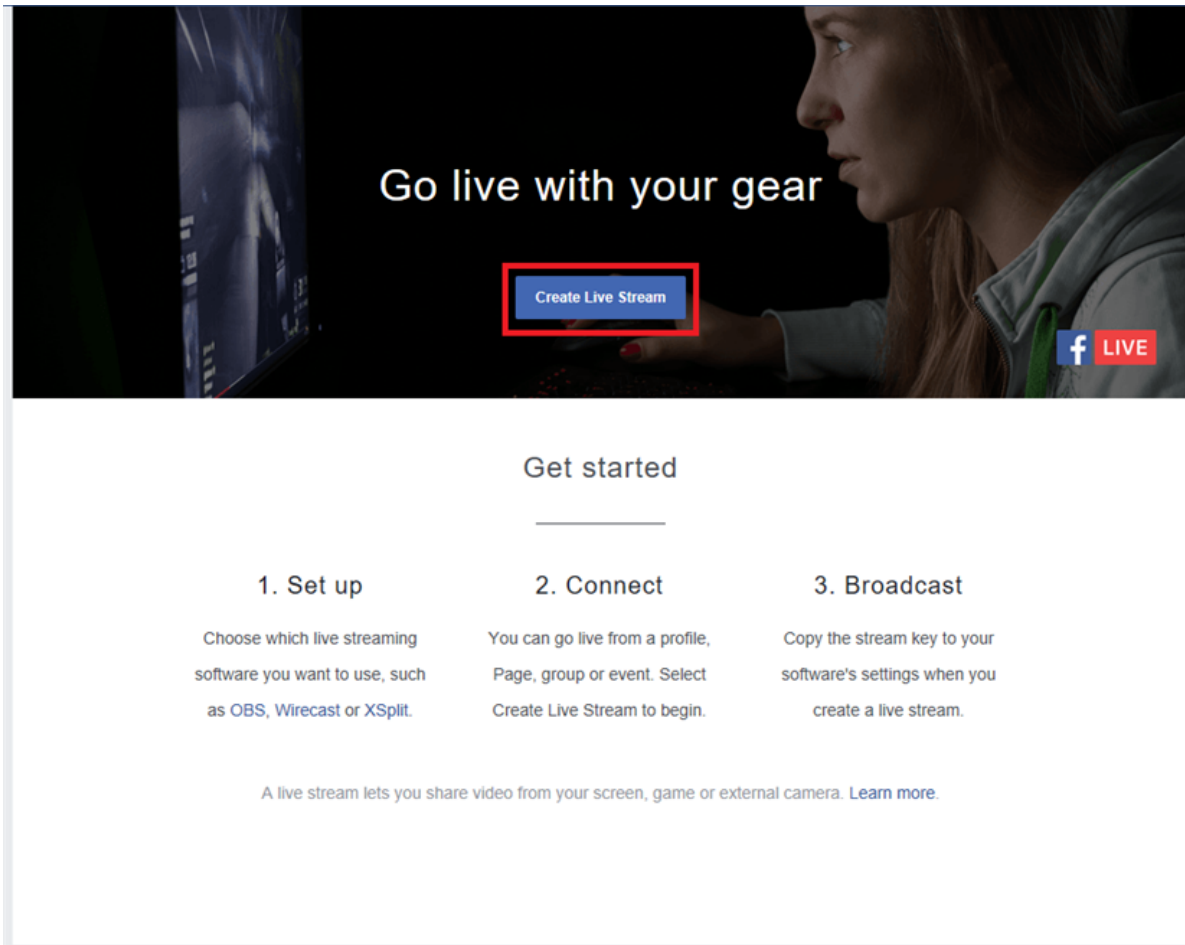
1. Ensure that the VMR you want to stream is configured with a **Maximum call rate** of **Full HD** (or uses a global default of Full HD).
2. Configure the outbound bandwidth on the VMR you want to stream to be 4096 kbps (**Maximum outbound call bandwidth** in the **Advanced options** of the VMR settings).
3. Ensure that the stream is capable of receiving 1080p. Within YouTube you must set up a **Reusable stream key** and configure the stream with a **Maximum sustained bitrate** of **3000 Kbps - 6000 Kbps (1080p)**.

Streaming a conference to Facebook

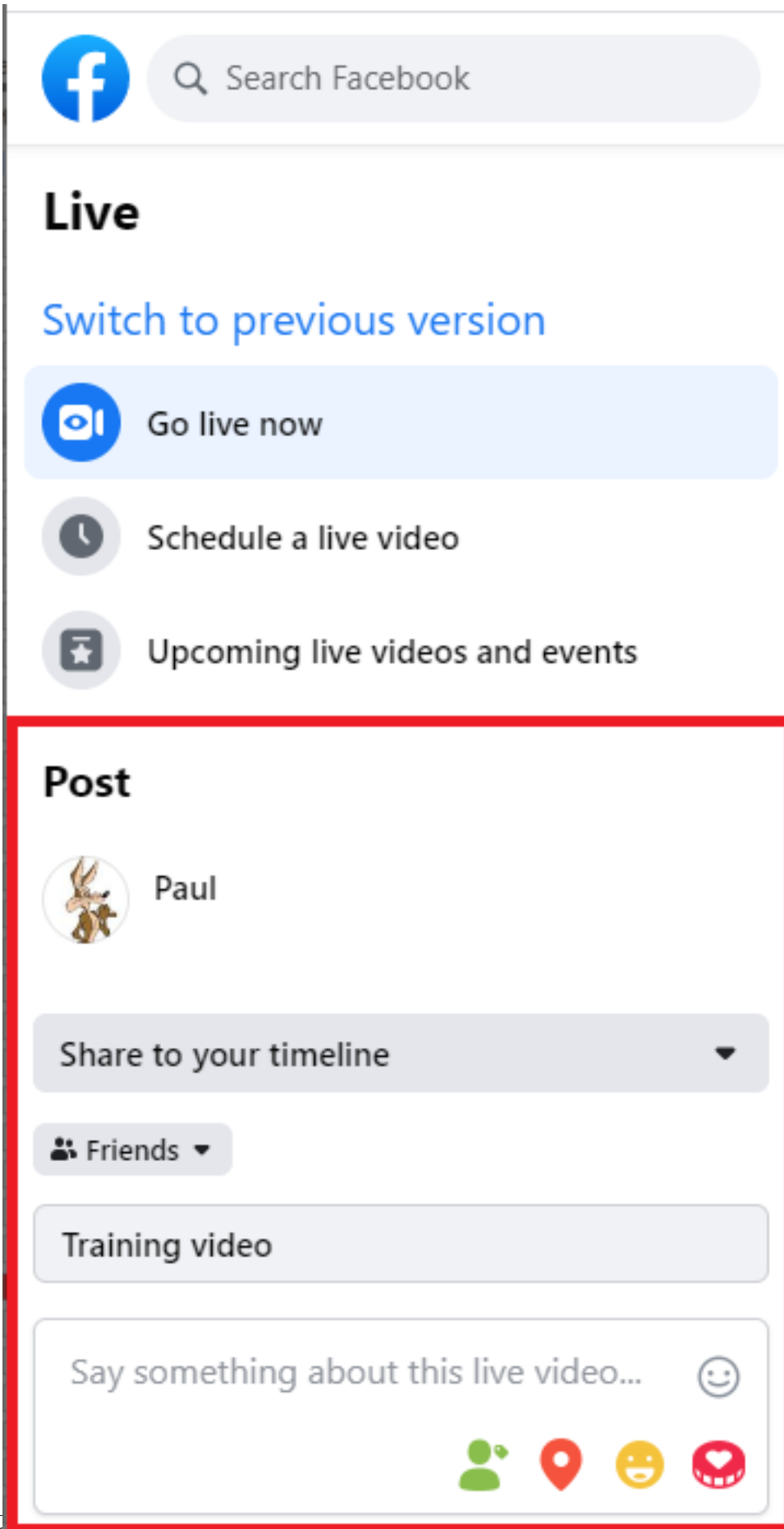
This guide explains how to stream a conference being held in a Pexip Virtual Meeting Room or Virtual Auditorium to Facebook.

To stream a conference to Facebook:

1. Go to facebook.com/live/create and select Create Live Stream:



2. From within the Post panel to the left of the page:
 - Select where to post your live broadcast (*Share to your timeline*, for example), and who should see it.
 - You can also add a Live video title and an associated message.



3. In the main body of the page, on the **Stream setup** tab, ensure that **Use stream key** is selected and that you can see the **Server URL** and **Stream key** settings. You will need these on the next step.

The screenshot shows the 'Stream setup' tab in the Pexip Infinity interface. The 'Get Started' section has three options: 'Use stream key' (selected and highlighted with a red box), 'Use camera', and 'Use paired encoder'. Below this, a note about pre-recorded content is visible. The 'Setup options' section has two checkboxes: 'Use a persistent stream key' and 'Use a backup stream', both of which are unchecked. The 'Settings' section has two dropdown menus: 'Stream' and 'Viewing'. The 'Live stream setup' section is highlighted with a red box and contains the following information:

- Server URL:** `rtmps://live-api-s.facebook.com:443/rtmp/` (with a 'Copy' button)
- Stream key:** `2670752349880486?s_bl=1&s_sc=2670752426547145&s_` (with a 'Copy' button)
- Event Logs** button

A 'Waiting for live video' button is located at the bottom right of the interface.

4. You must now initiate a call from the Virtual Meeting Room to Facebook, by adding the Facebook streaming address as a conference participant. You can do this either from the [Pexip Infinity Administrator interface](#) or from an [Infinity Connect client](#) connected to the VMR.

When using the Administrator interface, use the following settings:

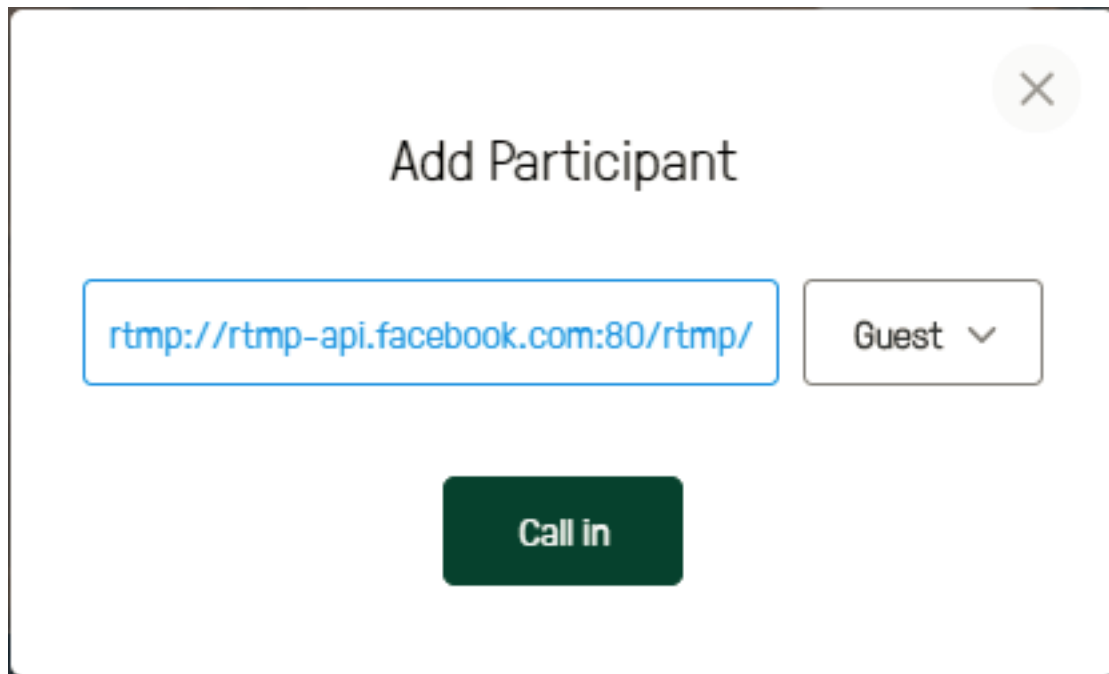
- **Protocol:** *RTMP*
- **Address:** the **Server URL** followed directly by the **Stream key**
- **Role:** we recommend selecting *Guest* (so that the streaming participant is not shown to other Guests in a Virtual Auditorium layout, and so that it does not keep a conference alive when all other Hosts have left).

When using an Infinity Connect client, use the following settings:



- **Participant details:** the **Server URL** followed directly by the **Stream key**, for example:
`rtmps://live-api-s.facebook.com:443/rtmp/2670752349880486?s_bl=1&s_sc=26707145&s_sw=0&s_vt=api-s&a=Abx7ECwitZM`

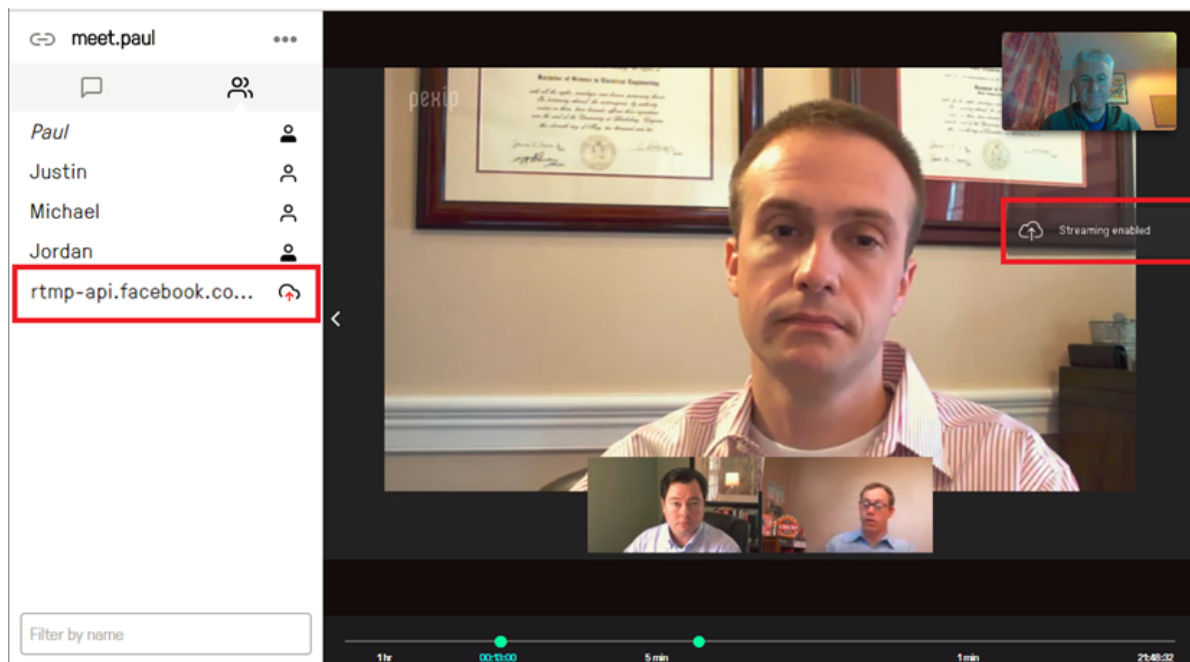
Note that a [suitable Call Routing Rule](#) is required when dialing out to a streaming service via Infinity Connect clients.

- Role: we recommend selecting *Guest*.



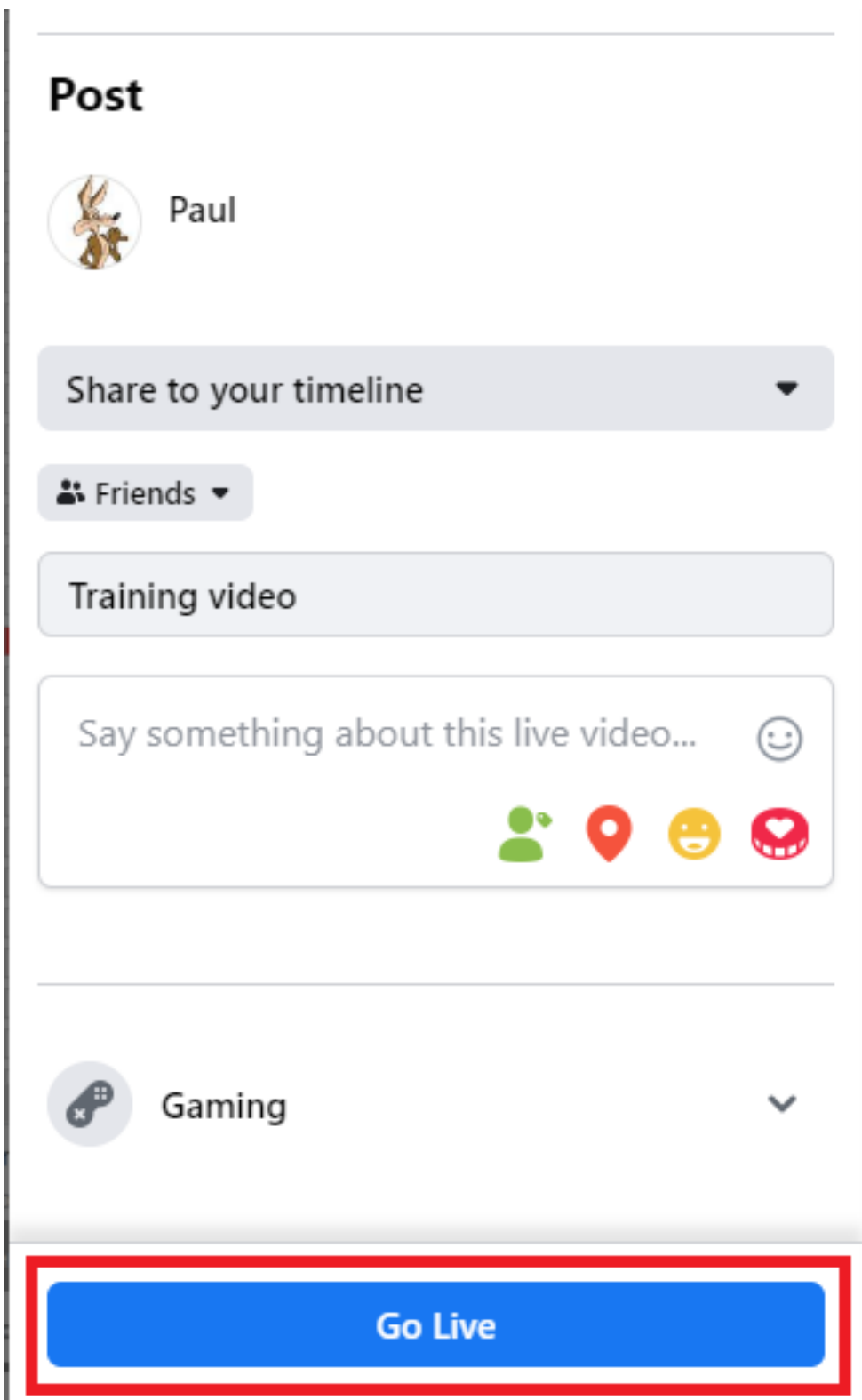
The 'Add Participant' dialog box features a title bar with a close button (X) in the top right corner. The main heading is 'Add Participant'. Below this, there is a text input field containing the RTMP URL 'rtmp://rtmp-api.facebook.com:80/rtmp/' and a dropdown menu currently set to 'Guest'. At the bottom center is a large green button labeled 'Call in'.

5. When Pexip Infinity has placed the call to the streaming service, the **Streaming enabled** icon  is displayed, and for Infinity Connect users the streaming participant appears in the participant list with a streaming badge  next to its name:

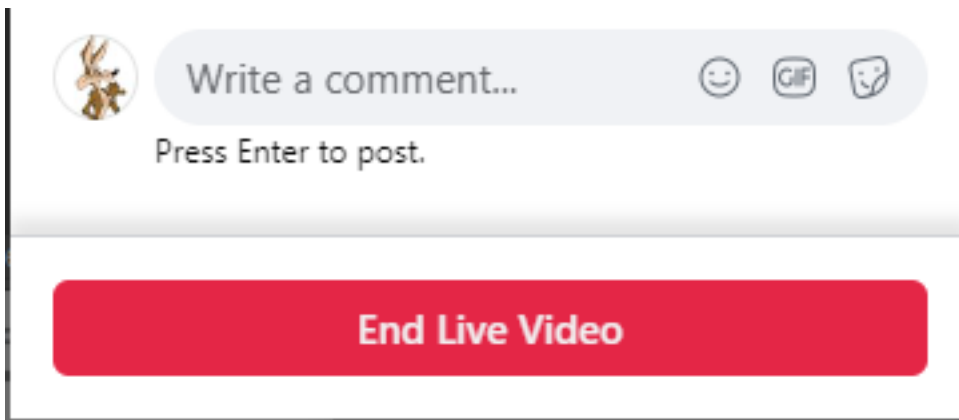


The screenshot shows the Pexip Infinity interface. On the left is a participant list for the meeting 'meet.paul'. The list includes names Paul, Justin, Michael, Jordan, and a streaming participant 'rtmp-api.facebook.co...' which is highlighted with a red box and has a streaming badge next to it. On the right is a video feed showing a man in a striped shirt. A red box in the top right corner of the video feed highlights the 'Streaming enabled' status. At the bottom, a timeline shows the meeting duration from 1hr to 26:48:32.

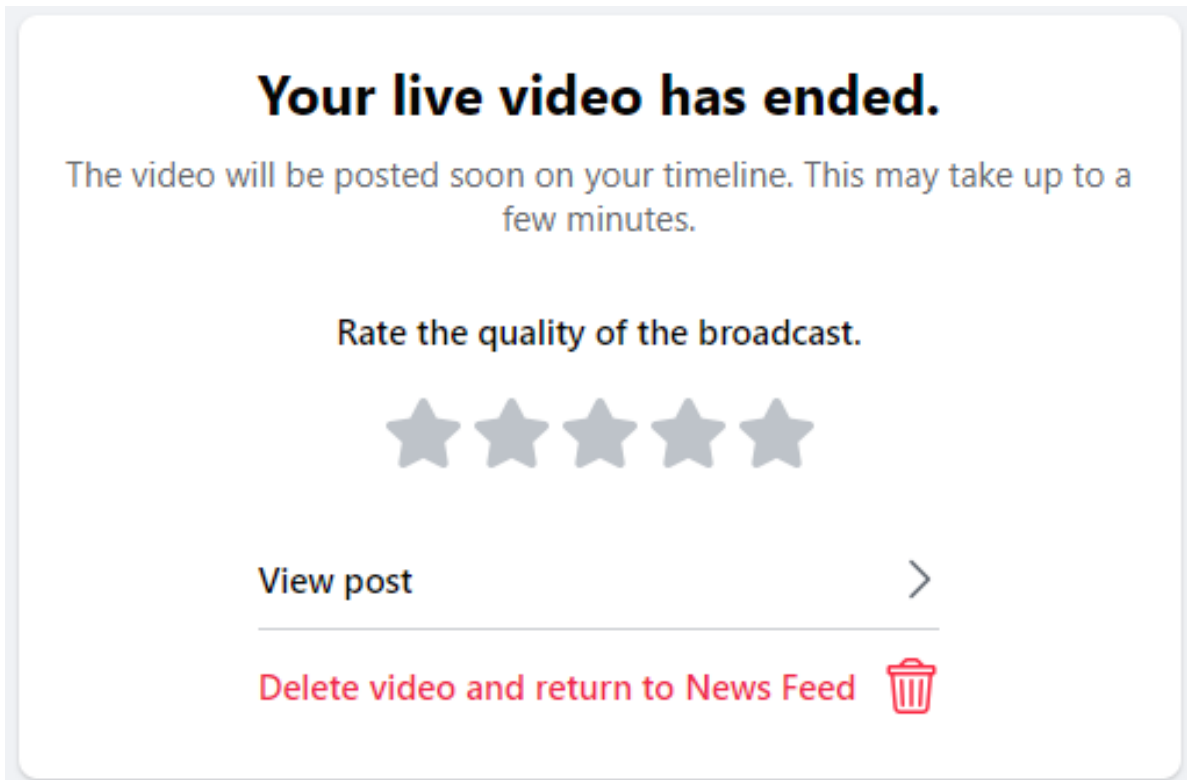
6. Return to Facebook. After a short delay you should see a preview of the video to be streamed. When you are ready, select **Go Live**:



7. You can see the live video and any comments as they are posted. When you have finished, select **End Live Video**:



8. Your video will be available on your timeline:



Streaming a conference to Periscope

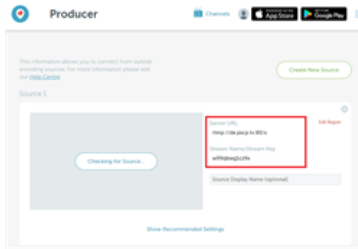
This guide explains how to stream a conference being held in a Pexip Virtual Meeting Room or Virtual Auditorium to Periscope.

To stream a conference to Periscope:

1. Go to <https://www.pscp.tv/account/producer> and select Create New Source > Normal Source:



2. You are given a **Server URL** and **Stream Name/Stream Key**:



3. Next you must initiate a call from the Virtual Meeting Room to the Periscope URL, by adding the Periscope URL as a conference participant. You can do this either from the [Pexip Infinity Administrator interface](#) or from an [Infinity Connect client](#) connected to the VMR. Alternatively, since Periscope URLs can be re-used for subsequent streams, you could set up the URL to be [automatically dialed](#) whenever a particular VMR is used.

When using the Administrator interface, use the following settings:

- **Protocol:** *RTMP*
- **Address:** the **Server URL**, followed by **/**, followed by the **Stream Name/Stream Key**
- **Role:** we recommend selecting *Guest* (so that the streaming participant is not shown to other Guests in a Virtual Auditorium layout, and so that it does not keep a conference alive when all other Hosts have left).

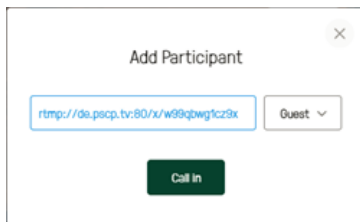
When using an Infinity Connect client, use the following settings:



- **Participant details:** the **Server URL**, followed by **/**, followed by the **Stream Name/Stream Key**, for example:

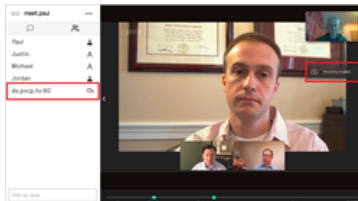
`rtmp://de.pscp.tv:80/x/w99qbwg1cz9x`

Note that a [suitable Call Routing Rule](#) is required when dialing out to a streaming service via Infinity Connect clients.

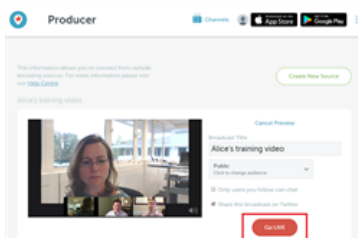
- **Role:** we recommend selecting *Guest*.



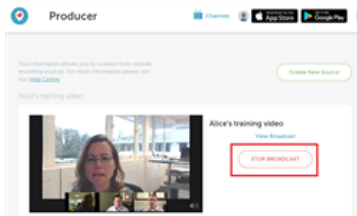
4. When Pexip Infinity has placed the call to the streaming service, the **Streaming enabled** icon  is displayed, and for Infinity Connect users the streaming participant appears in the participant list with a streaming badge  next to its name:



5. Return to Periscope and select **Preview Broadcast**. After a short delay you should see a preview of the video to be streamed. Add a title and select the viewing permissions. When you are ready, select **Go Live**:



- The video will now be available live to your followers. To view it and see and any comments as they are posted, select **View Broadcast**. This will open a new window with the URL for the broadcast, which you can then share and use to access the broadcast when it is finished.
- When you have finished, select **Stop Broadcast**.



Streaming a conference to Wowza Streaming Cloud

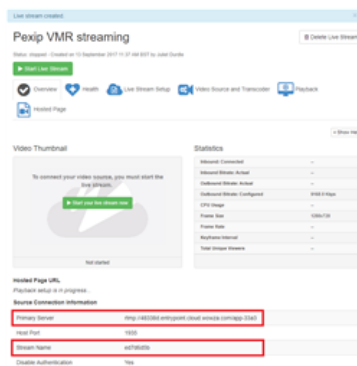
This guide explains how to stream a conference being held in a Pexip Virtual Meeting Room or Virtual Auditorium to Wowza Streaming Cloud.

To stream a conference to Wowza Streaming Cloud:

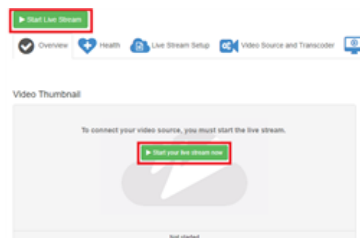
- Log in to your Wowza Streaming Cloud account at <https://cloud.wowza.com>.
- Go to **Live Streams > Add Live Stream**. You are taken through a series of pages asking you to provide information about the stream you are creating. Below are the settings you **must** select; for all other settings, choose the option most suited to your requirements:

Video Source and Transcoder settings	Option to select
What camera or encoder will you use to connect to Wowza Streaming Cloud?	Other RTMP
Do you want to push or pull your stream?	Push Stream
Source Security	Disable authentication

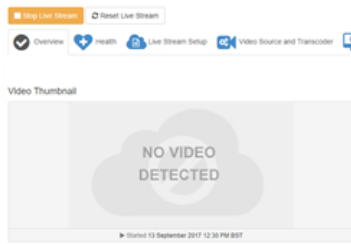
- Select **Finish**. You are taken to the **Overview** page for your new stream. In the **Source Connection Information** section you can see details of the stream's **Primary Server** and **Stream Name**:



- When you are ready to begin streaming, select **Start Live Stream**. You must begin the stream **before** you initiate the call from the VMR (see the following step).



When streaming has begun, you will see **No Video Detected** in the Video Thumbnail:



- Next you must initiate a call from the Virtual Meeting Room to the Wowza Streaming Cloud, by adding the Wowza stream as a conference participant. You can do this either from the [Pexip Infinity Administrator interface](#) or from an [Infinity Connect client](#) connected to the VMR. Alternatively, since Wowza streams can be re-used for subsequent events, you could set it up to be [automatically dialed](#) whenever a particular VMR is used.

When using the Administrator interface, use the following settings:

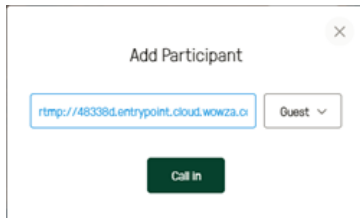
- Protocol: **RTMP**
- Address: the **Primary Server**, followed by **/**, followed by the **Stream Name**
- Role: we recommend selecting **Guest** (so that the streaming participant is not shown to other Guests in a Virtual Auditorium layout, and so that it does not keep a conference alive when all other Hosts have left).



When using an Infinity Connect client, use the following settings:

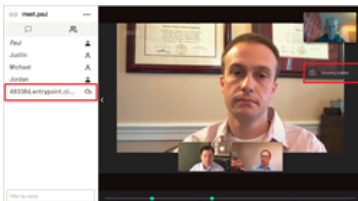
- Participant details: the **Primary Server**, followed by **/**, followed by the **Stream Name**, for example:
`rtmp://48338d.entripoint.cloud.wowza.com/app-33a3/ed7d6d5b`

Note that a [suitable Call Routing Rule](#) is required when dialing out to a streaming service via Infinity Connect clients.

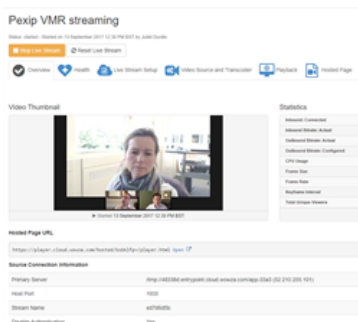
- Role: we recommend selecting **Guest**.



- When Pexip Infinity has placed the call to the streaming service, the **Streaming enabled** icon  is displayed, and for Infinity Connect users the streaming participant appears in the participant list with a streaming badge  next to its name:

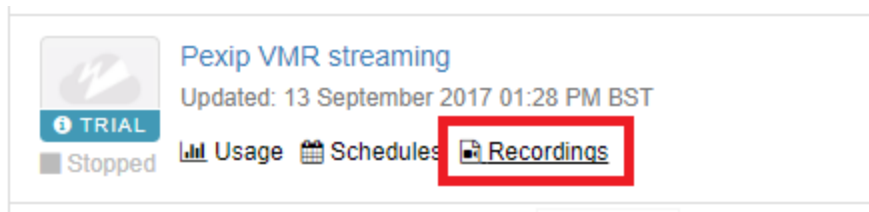


- Return to Wowza. After a short delay you should see a preview of the video being streamed:



- The stream is now available for viewing. If you chose the option to have Wowza Streaming Cloud host a webpage that plays back your video, anyone with the **Hosted Page URL** can view the live stream.

9. When you have finished, select **Stop Live Stream**.
10. If you have chosen to record the stream, after a short delay you can view and download this and any previous recordings of the stream by selecting the stream from the left-hand panel and then selecting **Recordings**.



Streaming a conference to Microsoft Stream

This guide explains how to stream a conference being held in a Pexip Virtual Meeting Room or Virtual Auditorium to Microsoft Stream (a component of Office 365).

Process overview

To stream a conference to MS Stream, you must:

- Have configured a suitable Call Routing Rule on Pexip Infinity that will take an ingest URL provided by MS Stream, append a suffix to it, and then connect out to Stream from an appropriate Conferencing Node.
- Create the live event and obtain an RTMPS (secure) ingest URL from Stream.
- Initiate a call from the conference to that Stream ingest URL, by adding the URL as a conference participant. You can use an Infinity Connect client or the Pexip Infinity Administrator interface to do this.

We recommend that you use a secure RTMP connection to Stream i.e. generate an `rtmps://` server ingest URL. This ensures that if you use an Infinity Connect client to add the `rtmps://` streaming participant URL, it will use Pexip Infinity's automatic call routing logic to place the call. Using a Call Routing Rule to process the streaming participant join request allows you to configure a suitable rule that will:

- Modify the ingest URL so that Pexip Infinity can use it to connect to Stream.
- Ensure that the call is placed from a node/location that can route the conference's stream to the Microsoft Stream RTMP ingest endpoint.

All of these steps are explained in more detail below.

Pexip Infinity Call Routing Rule configuration

When an Infinity Connect client or the Administrator interface places a call via automatic routing to the Stream **Server ingest URL**, it must be matched by a Call Routing Rule in Pexip Infinity that will adjust the **Server ingest URL** and then place the call from an appropriate Conferencing Node.

Modifying the server ingest URL

The **Server ingest URL** provided by Stream must be modified to enable it to work with Pexip Infinity. An additional folder name has to be appended to the end of the URL. This additional folder name suffix is an arbitrary string but we suggest `/msstream` is used.

For example, if your **Server ingest URL** obtained from Stream is `rtmps://xxx.channel.media.azure.net:nnnn/aaa` then the address that Pexip Infinity needs to use when dialing out to Stream must be something like `rtmps://xxx.channel.media.azure.net:nnnn/aaa/msstream`.

The person adding the streaming participant to the conference does not have to perform any modification to the ingest URL they copy from Stream — the modification of the URL is performed by the Call Routing Rule that is used to place the call to Stream, as described in our example rule below.

Configuring the rule

To create a suitable Call Routing Rule:

1. Go to **Services > Call Routing** and select **Add Call Routing Rule**.
2. The following table shows the fields to configure for your Call Routing Rule.

(Leave all other fields with default values or as required for your specific deployment.)

Option	Setting
Name	A suitable description such as "Streaming to MS Stream".
Priority	This is a very specific rule, so we recommend that you give it a relatively high priority (a low number).
Incoming gateway calls	If you want to use this rule just for calls placed from a conference, then leave this blank (disabled). However, if you also want to use this rule to enable incoming gateway calls to be placed out to MS Stream, then enable this option.
Outgoing calls from a conference	Enabled
Match against full alias URI	Enabled
Destination alias regex match	(rtmps?://[a-z0-9\-\+]\.channel\.media\.azure\.net(?:\d+)?/live/[a-z0-9\-\+])/?
Destination alias regex replace string	\1/msstream
Outgoing location	Ensure that you select a location that is able to place calls to MS Stream, such as a location containing Proxying Edge Nodes.
Protocol	RTMP (streaming)

Note that:

- The **Server ingest URL** generated by Stream always includes the `.channel.media.azure.net` domain which allows us to use a specific **Destination alias regex match** and thus minimize the chance of this rule conflicting with any other elements of your dial plan.
- The **Destination alias regex replace string** takes the dialed alias and appends `/msstream` which is necessary to ensure the dial out will work.

3. Select Save.

Now, when the meeting host adds the streaming participant and enters an alias in the format `rtmps://xxxx.channel.media.azure.net:nnnn/aaaa` it will be processed by the Call Routing Rule, transformed into `rtmps://xxxx.channel.media.azure.net:nnnn/aaaa/msstream`, and connect out over secure RTMP to Stream from an appropriate Conferencing Node.


Firewall requirements to enable calls to be placed to MS Stream

When using MS Stream you must ensure that your firewall allows outbound traffic from your Conferencing Node to TCP ports 1935/2935/1936/2936.

See <https://docs.microsoft.com/en-us/stream/network-overview> for more information.

Creating the live event in Stream and obtaining an RTMPS ingest URL

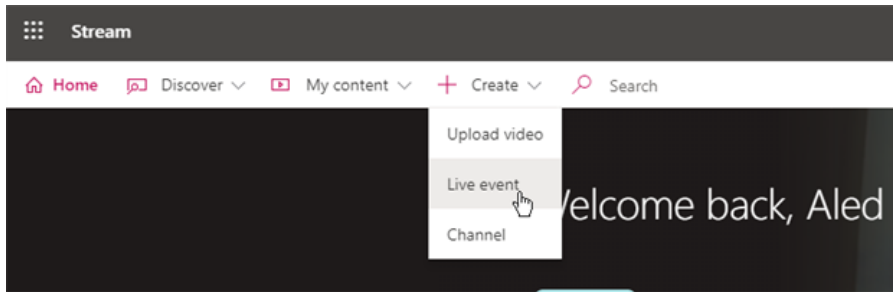
Within Microsoft Stream you can create a live event which will provide an RTMPS ingest URL.

-  Note that only certain users may have been granted the Live Event permission — it has to be given to them by the Microsoft Stream administrator.

For a detailed explanation of how to use and manage MS Stream, see <https://docs.microsoft.com/en-us/stream/live-create-event> for setting up a live event in your Office 365 instance.

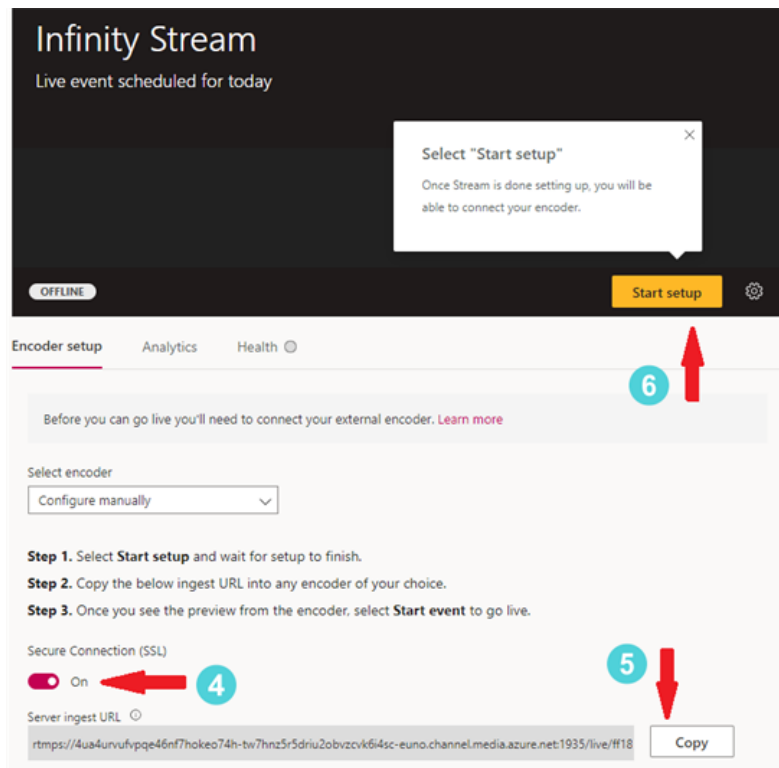
To create a live event:

1. From within Stream, go to Create > Live Event.



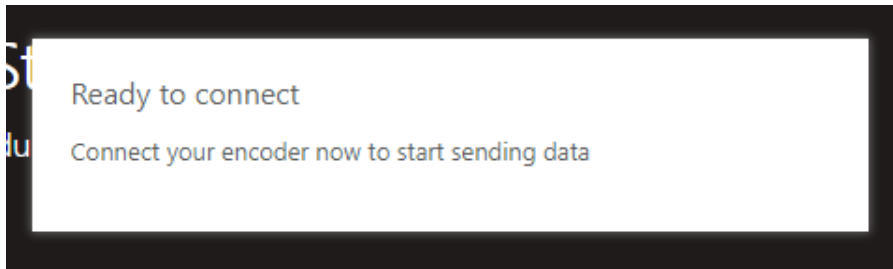
2. Enter all of the necessary event information such as a name, description, start time, permissions etc.
If you want to start streaming immediately, set the start time to *As soon as an encoder is connected (now)*.
3. Select Save.

The page then shows the Encoder setup where the Server ingest URL is displayed:



4. Set Secure Connection (SSL) to *On*.
5. Select *Copy* (next to the Server ingest URL) so that you can later paste the URL as the address you want to dial from the conference.
6. Select *Start setup*.

The live event then enters the pre-live stage. When Stream is ready to receive the RTMP stream from Pexip Infinity, you will see the message *Ready to connect*.



You can now initiate the conference connection from an Infinity Connect client or via the Pexip Infinity Administrator interface.

Connecting (dialing out) to Stream from Pexip Infinity

When you have obtained the **Server ingest URL** from Stream, you must then add that URL as a participant to the conference you want to stream. You can use an Infinity Connect client or the Pexip Infinity Administrator interface to do this. Note that the Pexip Infinity Call Routing Rule that we use to process the call will modify the URL slightly to enable Pexip Infinity to successfully dial out to Stream.

Adding the participant via an Infinity Connect client

When using an Infinity Connect client to add the streaming participant:

1. Join the conference in the normal way, or you can join as "Content" i.e. control-only if you are only using Infinity Connect as a means of initiating streaming.
2. From the toolbar at the bottom of the screen, select **Add participant**.
3. At the prompt, enter the **Server ingest URL** exactly as you copied it from Stream.
4. We recommend selecting a role of *Guest*.
5. Select **Call in**.

Adding the participant via the Pexip Infinity Administrator interface

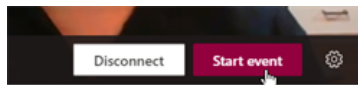
When using the Administrator interface to add the streaming participant, use the following settings for the new participant:

- **System location:** select a location that is able to place calls to MS Stream, such as a location containing Proxying Edge Nodes.
- **Participant alias:** the **Server ingest URL** exactly as you copied it from Stream.
- **Route this call:** *Automatically*
- **Role:** we recommend selecting *Guest*.
- **Streaming:** select this option.

Starting the stream

After you have successfully connected the RTMP stream from Pexip to the live event:

1. Return to the live event page in Stream where you will see a delayed video preview of the conference.
2. When you are happy that the stream is working then you can begin the live event by selecting **Start event**.



Note that the live stream will have a 20-30 second delay. This is because Stream buffers the stream so that it can tolerate brief connection losses and to ensure a good consistent experience. This is standard streaming behavior.

Stopping the stream

To stop the Live Event, from the MS Stream producer controls select **End Event**. It is best practice to stop the event in Stream before disconnecting your encoder (the stream from the Pexip VMR) otherwise audience members will see an error.

Troubleshooting

If the streaming participant fails to connect to Stream you must ensure that:

- The Conferencing Node placing the call is able to route calls to Stream. It needs to be able to route out to the internet and to TCP ports 1935/2935/1936/2936. The Conferencing Node placing the call is determined by the **Outgoing location** defined in the Call Routing Rule.
- The **Server ingest URL** entered as the alias to call is an `rtmps://` address and is the same address as shown in Stream. The Stream encoder setup must have **Secure Connection (SSL)** set to *On*. Otherwise, if an `rtmp://` address is used (i.e. SSL is off) and an Infinity Connect client is used to add the participant then the call will be placed directly to Stream and will not use a Call Routing Rule. This means that the call will fail because:
 - the arbitrary label has not been added to the end of the **Server ingest URL**, and/or
 - the call could not connect to Stream as the Conferencing Node placing the call (typically the node handling the connection to the Infinity Connect client) cannot route calls out to Stream.

Integrating with telephone systems (PSTN)

You may wish to enable standard PSTN and mobile telephones to dial in to your Pexip Infinity deployment. This allows callers using these devices to join a meeting as an audio-only participant.

To do this, you need to implement a trunk that will route your specified PSTN numbers to the aliases of your Virtual Receptions. Once in the Virtual Reception, participants can use their telephone keypad to enter the numeric alias of the Virtual Meeting Room or Virtual Auditorium they wish to join.

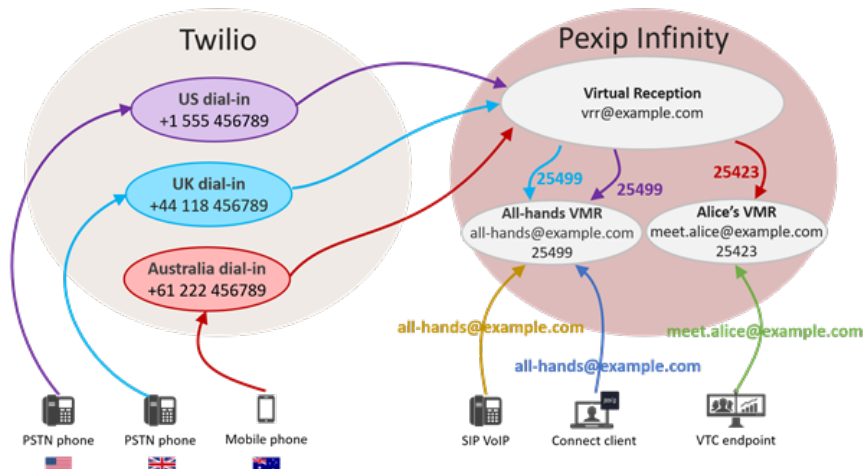
i You can also route PSTN numbers directly to Virtual Meeting Room aliases, but this is less common because it requires a 1:1 mapping of PSTN number to VMR, whereas using a Virtual Reception means that a single PSTN number can be used to access multiple VMRs.

You can use any service but in this topic we use Twilio as an example.

Overview

The steps required to enable PSTN access to VMRs are:

1. Obtain one or more PSTN numbers — usually one for each country, city or other geographical location from which you want to provide telephone access.
2. Configure the PSTN service so that calls to each number are routed to the alias of your Pexip Infinity Virtual Reception. Depending on the service you are using, you may need to configure the Virtual Reception with one alias per PSTN number, or you might be able to route all PSTN numbers to the same alias.
3. Ensure that each Virtual Meeting Room and Virtual Auditorium that you want to be accessible via PSTN has a numeric alias (in addition to any other aliases it may already have).



Overview of PSTN to Pexip Infinity integration

The diagram above gives an example of how PSTN calls can be routed to Pexip Infinity.

When telephone participants call their local PSTN number, they are routed to the Pexip Infinity Virtual Reception. From there, the participants enter the numeric alias of the Virtual Meeting Room they wish to join. At the same time, participants using other standards-based endpoints, including SIP VoIP telephones, SIP/H.323 VTC endpoints and Pexip Infinity Connect clients, can access the same Virtual Meeting Rooms directly by dialing the room's URI.

Prerequisites

To enable PSTN dialing to Virtual Meeting Rooms, you must ensure that:

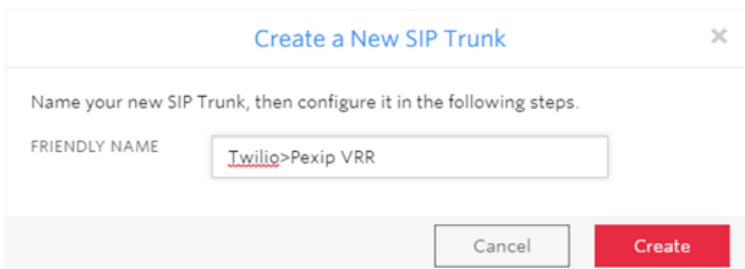
- Every Virtual Meeting Room and Virtual Auditorium that you wish to be accessible from the Virtual Reception has a numeric alias (in addition to any other aliases it already has). For more information, see [Configuring Virtual Meeting Rooms \(VMRs\)](#).
- You have appropriate Call Routing Rules in place. For more information, see [Configuring Call Routing Rules](#).
- You must have appropriate `_sips._tcp` DNS SRV records configured for the domain to be used in the alias of the Virtual Reception. In this example, we will be using the alias `vrr@example.com` so we need to have DNS records for `_sips._tcp.example.com` for each Conferencing Node in our deployment. For more information, see [DNS record examples](#).

Example using Twilio

In this example, we assume you have a Twilio account. Free trial accounts are available that offer limited functionality sufficient for testing integration with Pexip Infinity.

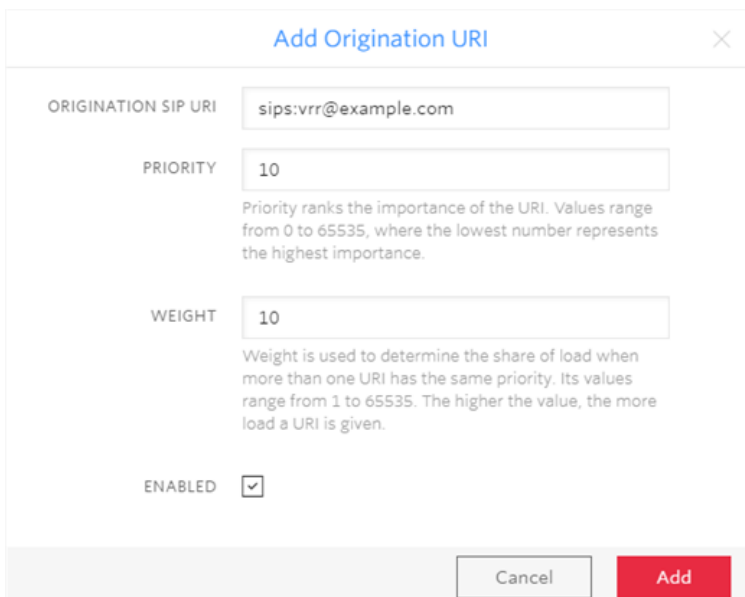
Step 1: Create a SIP trunk

1. From the Twilio interface, select the option to **Create new SIP Trunk**.
2. Give the trunk a name:



The screenshot shows a dialog box titled "Create a New SIP Trunk" with a close button (X) in the top right corner. Below the title, it says "Name your new SIP Trunk, then configure it in the following steps." There is a label "FRIENDLY NAME" followed by a text input field containing "Twilio>Pexip VRR". At the bottom, there are two buttons: "Cancel" and "Create".

3. Go to the **Origination** settings and select **Add new Origination URI**.
In the **Origination SIP URI** field, enter `sips:` followed by the alias that you will use for your Virtual Reception.
In our example, we enter `sips:vrr@example.com`



The screenshot shows a dialog box titled "Add Origination URI" with a close button (X) in the top right corner. It contains the following fields and controls:

- ORIGINATION SIP URI**: A text input field containing "sips:vrr@example.com".
- PRIORITY**: A text input field containing "10". Below it, a note states: "Priority ranks the importance of the URI. Values range from 0 to 65535, where the lowest number represents the highest importance."
- WEIGHT**: A text input field containing "10". Below it, a note states: "Weight is used to determine the share of load when more than one URI has the same priority. Its values range from 1 to 65535. The higher the value, the more load a URI is given."
- ENABLED**: A checkbox that is checked.

At the bottom, there are two buttons: "Cancel" and "Add".

Step 2: Assign PSTN numbers

1. Go to Numbers and select either **Add an Existing Number** (if you already have one configured) or **Buy a Number**.

Buy A Number - Search Results

NUMBER

Optional

Capabilities

Search

NUMBER	LOCATION	TYPE	CAPABILITIES	PRICE	
			VOICE SMS MMS	PER NUMBER	
+44 1757602057 (beta)	Selby, England	Local		\$1.00	Buy
+44 1174562539 (beta)	Bristol, England	Local		\$1.00	Buy
+44 8081694578	United Kingdom Proper	Tollfree		\$2.00	Buy
+44 8008085892 *	United Kingdom Proper	Tollfree		\$2.00	Buy
+44 8008085198 *	United Kingdom Proper	Tollfree		\$2.00	Buy
+44 8008021584 *	United Kingdom Proper	Tollfree		\$2.00	Buy
+44 8008021606 *	United Kingdom Proper	Tollfree		\$2.00	Buy
+44 3330165893	United Kingdom Proper	Local		\$1.00	Buy
+44 8008021373 *	United Kingdom Proper	Tollfree		\$2.00	Buy

← Revise Search

Refresh Results

2. Select the telephone number(s) you wish to assign, and then **Add Selected**.

In this example, we have set up the following PSTN numbers for our SIP trunk:

- US: +1 555 456789
- UK: +44 118 456789
- Australia: +61 222 456789

Step 3: Create a Virtual Reception

We create a Virtual Reception and assign it the alias that we've used for the SIP trunk, as follows:

1. From the Pexip Infinity Administrator interface, go to **Services > Virtual Receptions** and select **Add Virtual Reception**.
2. Give the Virtual Reception a **Name** and an **Alias**. The **Virtual Reception type** should be left at the default **Regular**. (For more information about configuring Virtual Receptions, see [Configuring Virtual Reception IVRs](#).) Then add each of the aliases you have already configured in Twilio:

Option	Input	Notes
Name	Toll-free Virtual Reception access from Twilio	This single Virtual Reception will be accessible globally by local toll-free numbers.
Description	Allow users to dial a local telephone number and then select a VMR.	
Service options		
Virtual Reception type	<i>Regular</i>	
Aliases		
Alias: #1		
Alias	vrr@example.com	
Description	SIP trunk from Twilio	

Now whenever participants dial any of the PSTN numbers set up in Step 2, they will be taken to the same central Virtual Reception.

Step 4: Add numeric aliases to Virtual Meeting Rooms and Virtual Auditoriums

Next, we need to ensure that our Virtual Meeting Rooms and Virtual Auditoriums are accessible from the Virtual Reception. To do this, we add a new, unique numeric alias to each Virtual Meeting Room and Virtual Auditorium. In this example, we want participants to be able to access Alice's Virtual Meeting Room by entering **25423** on their keypad, and the All Hands Virtual Auditorium by entering **25499**, so we edit these services as follows:

Alice's VMR

Option	Input	Notes
Name	Alice's VMR	
Description	Alice's personal meeting room	
Aliases		
Alias: #1		
Alias	meet.alice.jones@example.com	This is the existing alias for Alice's VMR.
Description	URI for Alice's VMR	
Alias: #2		
Alias	25423	We add this new alias to the existing VMR configuration.
Description	Number for Alice's VMR when accessing via the Virtual Reception.	

All Hands Virtual Auditorium

Option	Input	Notes
Name	All Hands	
Description	Virtual Auditorium for All Hands conference	
Aliases		
Alias: #1		
Alias	all-hands@example.com	This is the existing alias for the All Hands conferences.
Description	URI for All Hands conference	
Alias: #2		
Alias	25499	We add this new alias to the existing Virtual Auditorium configuration.
Description	Number for All Hands when accessing via the Virtual Reception.	

Joining a Virtual Meeting Room

Now Alice can tell participants to join her VMR by doing any of the following:

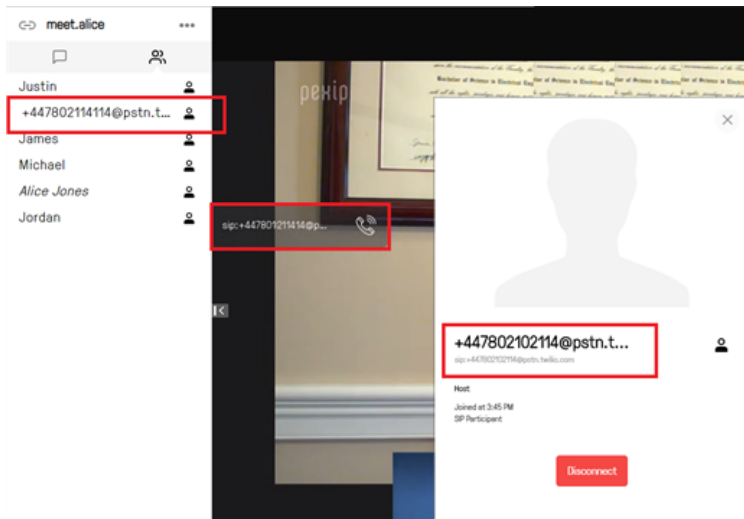
- dialing **meet.alice.jones@example.com**
- from the US, dialing **+1 555 456789** and then entering **25423**
- from the UK, dialing **+44 118 456789** and then entering **25423**
- from Australia, dialing **+61 222 456789** and then entering **25423**.

Likewise, participants can join an All Hands conference by doing any of the following:

- dialing **all-hands@example.com**
- from the US, dialing **+1 555 456789** and then entering **25499**
- from the UK, dialing **+44 118 456789** and then entering **25499**
- from Australia, dialing **+61 222 456789** and then entering **25499**.

So in our example, a telephone participant in the US dials **+1 555 456789** and is routed to the Pexip Infinity Virtual Reception. There they hear an audio prompt asking them to enter the number they wish to connect to. They enter the numeric alias for Alice's VMR, **25499**, and join the VMR as an audio-only participant.

Other participants using an Infinity Connect client see the telephone participant in the roster with a name in the format **<participant's telephone number>@ptsn.twilio.com**:



Pexip Infinity diagnostics

For information about how to monitor and troubleshoot Pexip Infinity, see:

Viewing live and historical platform status	473
Conference status	480
Diagnostics tools and reporting	503
Viewing Conferencing Nodes	514
Viewing system location status	516
Viewing cloud bursting status	517
Viewing alarms	518
Viewing login history	528
About the support log	529
About the administrator log	531
Log output	533
Creating and viewing diagnostic graphs	546
Disconnection reasons	548
Pexip Infinity port usage and firewall guidance	552
Troubleshooting the Pexip Infinity platform	560

Viewing live and historical platform status

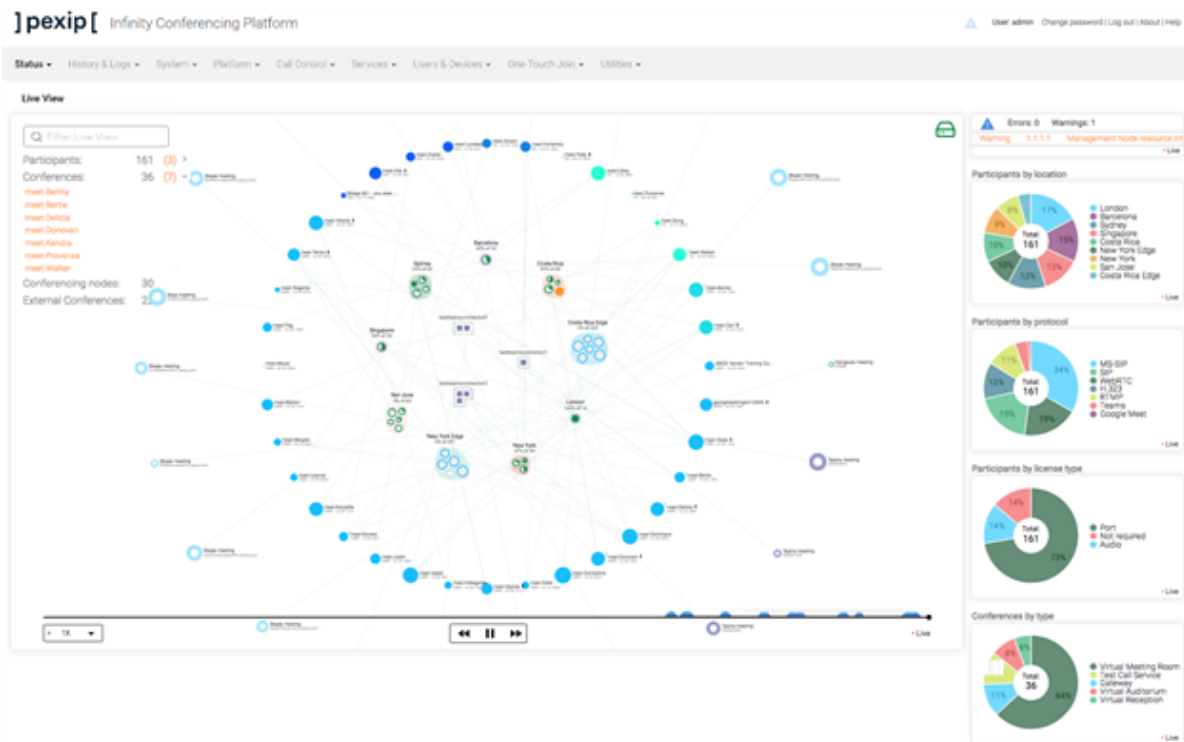
For a real-time overview of your Pexip Infinity deployment, go to **Status > Live View**. The interactive graphs and charts on this page show at a glance:

- the locations and Conferencing Nodes that are currently deployed and the available capacity and current load on each node
- all conferences that are currently taking place, and the nodes on which they are being hosted
- any conferences or conference participants that are experiencing call quality issues
- any error or warning alarms
- (during an upgrade process) which nodes are currently being upgraded, which nodes are still waiting to be upgraded, and which are in maintenance mode
- pie charts showing a breakdown of participants by location, protocol, license type and the different conference types being hosted.

You can interact with the graphs and charts by hovering your mouse pointer over each item to view specific information on each location, node and conference. You can also drill down to view more detailed current and historical information.

You can [filter](#) the view to show specific conferences or participants.















By using the timeline controls, you can [rewind and replay](#) the graph to view full node status and conference activities during the previous seven days. For a complete history of all conferences, see [Viewing historical information about conferences](#).



Key to icons and symbols

The status graphs use the following symbols:




Icon	Meaning	Icon	Meaning
Management Node			

Icon	Meaning	Icon	Meaning
	The Management Node in normal operation.		The Management Node with an error-level alarm.
	The Management Node with a warning-level alarm.	(Pulsating)	
(Pulsating)			
Conferencing Nodes *			
	A Transcoding Conferencing Node during normal operation. The amount of green fill within the circle indicates the current media load (in terms of percentage of estimated HD ports in use), so an unused node is white and a fully loaded node is filled entirely green.		A Proxying Edge Node during normal operation. The amount of green fill within the circle indicates the current media load (in terms of percentage of proxying capacity in use), so an unused node is white and a fully loaded node is filled entirely green.
			
			
			
	A cloud bursting node that is currently on standby i.e. stopped.		A node that is in <u>maintenance mode</u> .
	A cloud bursting node that is currently starting (green circle spinning clockwise) or stopping (spinning counter-clockwise).		A node that is waiting to be upgraded. When you begin an upgrade of your platform, all nodes will have this status but will still be able to handle calls. After each node is upgraded in turn, it will return to normal operation.
(Rotating)			
	A node that has an error-level alarm.		A node that is currently upgrading.
(Pulsating)		(Pulsating)	
	A node that has a warning-level alarm.		
(Pulsating)			
Teams Connectors			
	When enhanced Teams status is not enabled, Teams Connector instances are only displayed when they are currently involved in a Teams call. Each purple square represents a Teams Connector instance.		When enhanced Teams status is enabled, all Teams Connectors that are integrated with Pexip Infinity are shown. Each purple square represents a Teams Connector instance and the fill level of the square represents the current media load.
Conferences †			
	Conference being held in a Virtual Meeting Room.		Gateway call.
	Conference being held in a Virtual Auditorium.		Call to the Test Call Service.

Icon	Meaning	Icon	Meaning
	Call to a Virtual Reception.		A Skype for Business meeting.
	A conference that is experiencing call quality issues displays a blue asterisk next to the conference icon.		A Microsoft Teams meeting.
	Conference is locked.		A Google Meet meeting.

The labels shown against each conference are based on the name of the VMR, Virtual Auditorium and so on, or the Call Routing Rule name for gateway calls.

Participants

	The participant is currently presenting content.		API participant.
	Streaming participant.		

* For Conferencing Nodes, the size of each icon is relative to its total capacity.

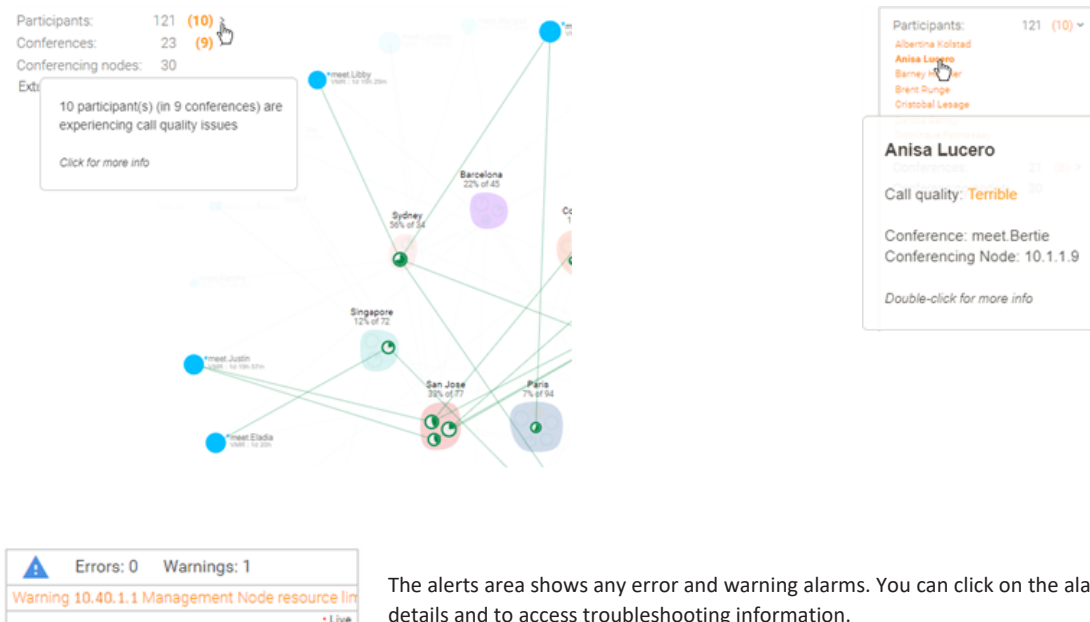
† For conferences, the size of each icon is relative to the number of participants.

Platform summary status, call quality issues and alarms

The statistics at the top of the graph summarize the platform status. They show the number of participants currently connected to the platform, the number of concurrent conferences, the number of Conferencing Nodes, and the number of externally-hosted conferences (i.e. Microsoft Teams or Skype for Business meetings, or Google Meet).

The issues statistics (in orange) are only shown if any of the participants or individual conferences are experiencing call quality issues. If you hover over this area, the affected participants and conferences are indicated.

You can click to expand the list of issues and you can also review the details of individual participants and conferences.



The alerts area shows any error and warning alarms. You can click on the alarm for more details and to access troubleshooting information.

The interactive timeline indicates in blue any times in the past when a participant or backplane had call quality issues. You can hover over each blue bubble to get details of the issues that occurred at that time. In very large/busy systems with poor networks this is limited to the last 10,000 events over the last 7 days.

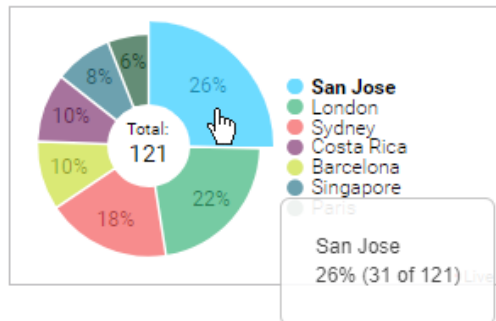
Pie charts and detailed participant usage graphs

The pie charts show a breakdown of:

- Participants by location
- Participants by protocol
- Participants by call license type (port, audio, or not required) — note that this shows total participants; some types of participant calls consume 2 licenses
- Conference types (VMRs, Virtual Receptions, gateway calls etc.)

You can hover over an area of the chart to see more information.

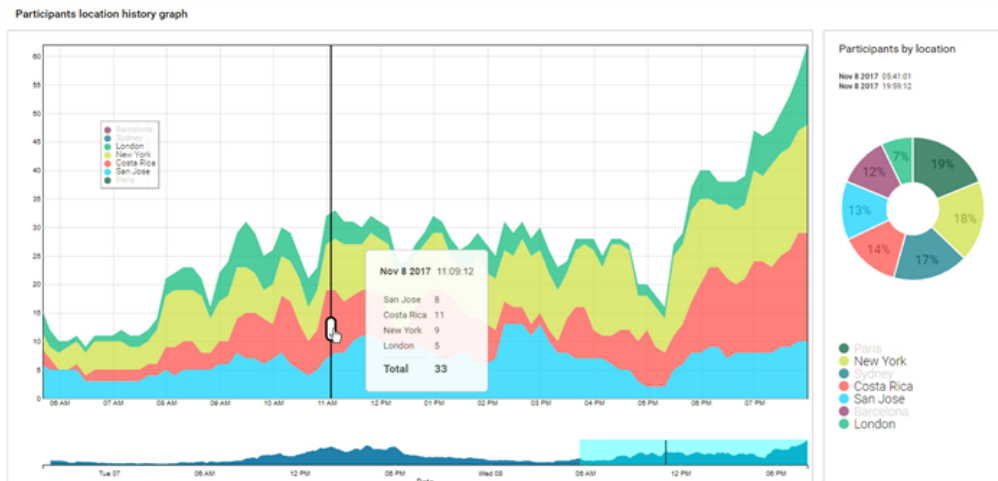
Participants by location



If you click on a pie chart you are taken to a more detailed interactive graph showing historical information about those locations, protocols, licenses or conference types, where you can filter out or select specific elements e.g. you could choose to only view participants using the SIP protocol, or all participants except those in the London location.

This example here shows the participants by location history graph, but all of the other graphs (protocol, license, conference type) work in the same way. You can:

- Click on individual locations (or protocols, licenses, conference types) to include/exclude them from the chart.
- Grab and move the vertical timebar up or down to adjust the time period that is displayed, or move the bar left or right to show the breakdown of participants for that point in time, or just click anywhere on the graph to move the vertical timebar to that point. The historical data is captured at 10 minute intervals.
- Adjust, or grab and move the timebox area in the bottom graph to quickly change the date and time period that is displayed in



detail in the main chart. You can double-click on the bottom graph to reset the main chart to zoom out to show all available data.

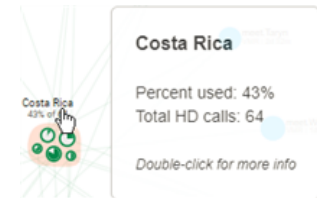
- Grab and move the legend to another area of the chart.

Viewing location status

Each location is represented by a group of one or more Conferencing Nodes against a colored background. A different color is used for each location. The label gives the name of the location and summarizes the amount of available HD port capacity across all Conferencing Nodes in that location, and shows how much of that capacity is currently in use.

You can double-click to get more detailed [location information](#), including an interactive status chart showing the load history for that location.

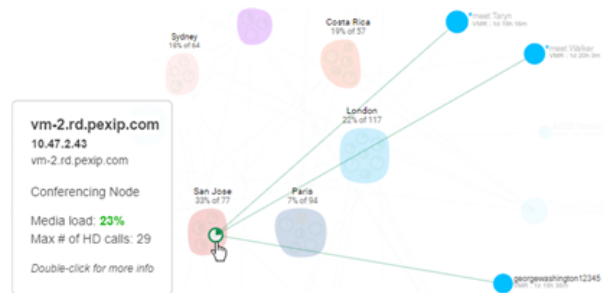
The example here shows the Costa Rica location, which is current using 43% of the 64 available HD ports.



Viewing Conferencing Node status

Each Conferencing Node is represented by a node [icon](#) . Nodes in the same location are grouped together. Hovering over each node's icon provides information about that node's hostname, IP address, current load and estimated capacity (in terms of the estimated maximum number of calls the node can handle). Hovering also highlights all the conferences for which that node is handling the media.

You can double-click to get more detailed [node information](#), including an interactive status chart showing the load history for that node.



The example here shows the information that is available for one of the four Transcoding Conferencing Nodes in the San Jose location.

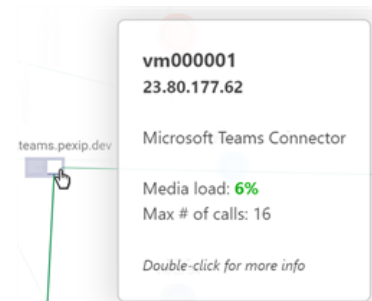
Viewing Teams Connector and Teams meeting status

You can view the status of each Teams Connector instance, such as call capacity and current media load, when you hover over an instance , providing enhanced Teams status information is enabled.


If enhanced status information is not enabled then Live View only displays Teams Connector instances when a Teams call is in progress.

Each purple square represents a Teams Connector instance. A single instance can handle more than one Teams meeting; if you hover over an instance you are shown all of the Teams meetings that it is currently running.

You can also hover over an individual Teams meeting to see a list of all the participants, and highlight those who are gatewayed into the meeting.

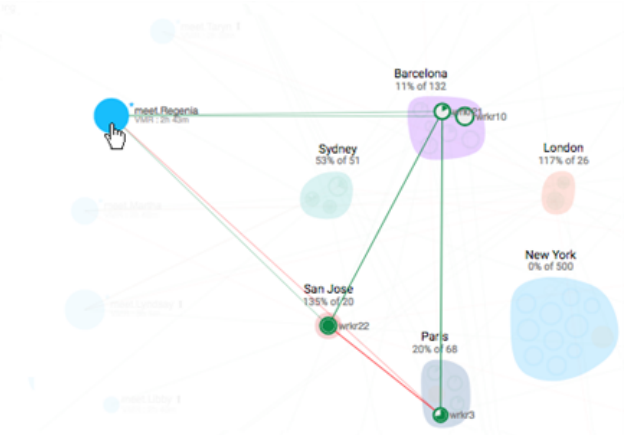


Viewing conference status

Conferences are represented by  icons. Hovering over the icon provides details of the conference, including its duration and a list of participants.

Hovering also highlights all the Conferencing Nodes that are handling media for that conference. Any backplane issues are highlighted — if a backplane link has poor quality, it is shown in red, and if a participant has poor quality, the link between their node and the conference is also shown in red.

meet.Regenia
 Virtual Meeting Room (VMR)
 Duration: 2h 43m
 Backplanes with issues: 1
 • 10.1.1.22 <-> 10.1.1.3
 Participants:
 • Arden Stalvey
 • Dewitt Jarrells
 • Eboni Hillery
 • Olinda Marmol
 • Shelton Ziglar
 • Tiana Carruthers
 • Valerie Almanza
 Double-click for more info



You can double-click to get [full conference details](#) and to view an interactive conference graph.

The example here shows the information that is available for the conference being hosted in the meet.Regenia Virtual Meeting Room. The conference is experiencing one backplane issue.

Gateway calls to externally-hosted conferences

When viewing a Gateway call to an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet, all of the externally-connected participants are listed (grayed out).

When you hover over the external conference it also shows you which Conferencing Node it's connected to, and a link to the associated Pexip gateway call.

Connecting to Google Meet
meeting:a0165e17-4c39-41fa-b686-e9d673c7f333
 Gateway (GW)

Duration: 38s

Participants:

- Alvaro
- Google Meet (825611141)
 - Alice
 - Bob
 - Paul

Double-click for more info

Filtering by participant or conference

The box at the top of the graph allows you to filter the view by conference or participant name.

When you enter text in the filter box, you will see only:

- Conferences with the text in their name,
- Conferences with the text in their service tag,
- Conferences which contain a participant with the text in their name, and
- Conferences which contain a participant with the text in their alias.

Live View

Q steve

Participants: 1 of 31
 Conferences: 1 of 2
 Conferencing nodes: 12

You can also apply a filter to an individual conference graph.

When a filter is applied in live view, the timeline at the bottom of the page indicates in yellow all the times in the past when there was a conference or participant matching the filter. If you hover over these yellow indicators, information about the match will appear.

If you select a conference to view while the filter is applied, or you apply a filter to conference graph, any participants whose name or alias matches the filter text are highlighted in the conference graph. When viewing a conference graph with a filter applied, the timeline indicates in yellow whenever there was a participant in the conference who matched the filter. Again, if you hover over these yellow indicators, information about the match will appear.

Search matches:

Participants:

- Steve

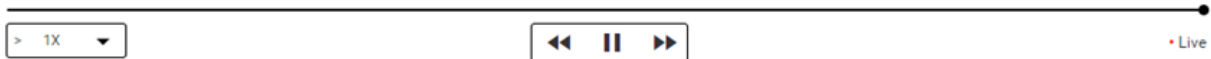
Jul 25 2019 10:22:21 (Europe/London)

Rewinding and replaying status

You can use the controls at the bottom of the graph to review the platform status during the previous seven days (data persists across reboots and upgrades). Any times when a participant or backplane had call quality issues are indicated in blue. You can use the timeline to:

- View historical Conferencing Node activities such as nodes being added or placed into maintenance mode.
- View conferences and the nodes on which they are hosted.
- Drill down into individual conferences to review conference activities such as participants joining, leaving or presenting, and examine a participant's media statistics.
- Pause, rewind and replay the graph at a variety of speeds.

The pie charts and alarms list also reflect the selected time period.



The following controls are available:

Control	Description
	Use the control at the bottom left of the timeline to select the speed at which to rewind or replay the graph.
	Use the control below the middle of the timeline to rewind, fast-forward, play or pause the graph.
	Drag the black dot to view a particular point in the timeline. The selected date and time are shown above the timeline.
	When viewing historical information, the date and time of the graph currently being displayed is shown at the bottom right of the timeline.
	To return to the live view, drag the black dot to the far right of the timeline, or fast-forward to the end. When you are viewing the live graph, "Live" will be shown at the bottom right of the timeline.

To maintain optimum system performance, the timeline is not available for large deployments of more than 40 Conferencing Nodes.

Conference status

Viewing current conference status

To see a list of all the conferences currently in progress, go to **Status > Conferences**. You can select individual conferences to see more detailed information, including an interactive conference [graph](#), and perform some [conference control](#) functions such as muting all Guests or disconnecting all participants.

To view historical information on conferences after they have finished, including the ability to rewind and replay the conference graph, see [Viewing historical information about conferences](#).

The pages showing conference status information do not refresh automatically — you must refresh them manually — except for the conference graph which does dynamically refresh. A "Conference instance no longer active" message indicates that the conference whose details you were viewing has now finished.

The Administrator interface uses color coding when reporting media statistics, such as perceived call quality, packet loss and jitter. In general, statistics that are shown in green represent good quality, orange represents intermediate quality, and red is used for bad quality. See [media statistics and perceived call quality](#) for more information.

Each conference has the following information available:

Field	Description
Service name	<p>The name of the Virtual Meeting Room, Virtual Auditorium, Virtual Reception, Test Call Service or Call Routing Rule. For Infinity Gateway calls, the rule name is followed by a unique identifier to distinguish between separate calls.</p> <p>For Virtual Receptions and Test Call Services, if there are multiple concurrent users of that service you will see a single instance of that service (rather than one instance per participant, as all participants are using the same service even though they cannot see or hear each other).</p> <p>Click on the service name to view more information.</p>
Duration	<p>For Virtual Meeting Rooms and Virtual Auditoriums: the length of time since the first participant joined the conference.</p> <p>For Virtual Receptions and Test Call Services: the length of time that this instance of the service has been in continuous use (note that this could involve more than one participant if their usage overlapped).</p> <p>For Infinity Gateway calls: the length of time since the call was received by the Infinity Gateway.</p>
Participant count	The number of participants currently in the conference or using the service.
Service type *	The type of conference, e.g. Virtual Meeting Room.
Service tag *	The unique identifier that an administrator has assigned to this service. If this field is blank, no tag has been assigned. For more information, see Tracking usage via service and participant call tags .
Is locked *	Indicates whether a conference has been locked to prevent further participants from joining. For more information, see Locking a conference and allowing participants to join a locked conference .
Guests muted *	Indicates whether all Guest participants are muted.
* Only displayed when you have selected an individual conference to view.	

To view more information about the conference, click on the service name. This gives you access to some [conference control](#) functions, and 3 tabs: [Participants](#), [Backplanes](#) and [Graph](#).

Conference control

When viewing conference details you can use the controls at the bottom of the page to:

- dial out to a new participant
- mute all Guests

- lock the conference
- disconnect all participants from the conference.

Participants

The **Participants** section lists all the participants that are currently in the conference.

For more details about a particular participant, including media stream statistics, click on the **Participant alias**. This takes you to the [Participant details](#) page.

Field	Description
Participant alias	The name of the user or the registered alias of the endpoint. When viewing a Google Meet conference, you will see additional aliases (typically in the form "spaces/<id>/devices/<id>") for each external participant.
Duration	The length of time since this participant joined the conference or accessed the service.
Display name	The name that has been configured on the participant's endpoint.
System location	The system location of the Conferencing Node to which the endpoint is connected. However, when the participant is connected to a Proxying Edge Node, this is the location of the Transcoding Conferencing Node that is processing the conference media for this participant.
Signaling node	The IP address and name of the Conferencing Node to which the endpoint is connected. This node is handling the call signaling but may or may not be handling the call media (for more information, see Handling of media and signaling in locally distributed conferences).
Media node	The IP address and name of the Transcoding Conferencing Node that is processing the call media for this participant (for more information, see Handling of media and signaling in locally distributed conferences).
Role	<p>Host indicates that either:</p> <ul style="list-style-type: none">• the conference has no PINs configured (in which case all participants have a role of Host)• the participant accessed the conference using the Host PIN. <p>Guest indicates that the participant accessed the conference using the Guest PIN.</p> <p>Unknown indicates one of the following:</p> <ul style="list-style-type: none">• the participant is at the PIN entry screen and has not yet successfully entered a PIN• the participant is at the Waiting for Host screen but their role has not yet been determined• the participant is connected to an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet (these participants appear in the Infinity Connect participant list with a role of External Guest). <p>For more information, see About PINs, Hosts and Guests.</p>
Is presenting	Indicates whether the endpoint is currently sending a presentation stream.
Is muted	Indicates whether the endpoint's audio has been muted (using an Infinity Connect client, the Administrator interface, or by a third party using the Pexip API).
On hold	Indicates whether the endpoint has been put on hold (usually by the endpoint user).
Streaming or recording device	Indicates if the participant is a streaming or recording device.

Backplanes

The **Backplanes** section provides information about the media streams being transmitted between Transcoding Conferencing Nodes for the selected conference. Backplane links between Conferencing Nodes are unidirectional, so for a conference involving two transcoding nodes there will be two backplane links: one from node A to node B, and another from node B to node A. Note that a bidirectional backplane is created when a Conferencing Node connects to a Teams Connector or to a Skype for Business / Lync meeting.

Field	Description
Media node	The IP address and name of the Conferencing Node that is transmitting media. For details about the media streams being sent over a particular backplane link, click on the media node's IP address.
Remote media node	The IP address and name of the Conferencing Node or remote system e.g. a Teams Connector, that is receiving media. Note that the remote media node of a merged SfB/Lync meeting is identified by the address of the SfB/Lync client that initiated the SfB/Lync meeting.
Remote conference name *	The name of the conference on the remote node. For external backplanes, this identifies the conference on the other platform, such as a Microsoft Teams conference ID.
System location *	The system location of the Conferencing Node that is transmitting media.
Duration	The length of time since the connection was established.
Backplane type	Geographic indicates that the two Conferencing Nodes are in different system locations . Local indicates that the two Conferencing Nodes are in the same system location . External indicates a link between a Conferencing Node and an external node, such as a Teams Connector.
* Only displayed when you have selected an individual media node to view.	

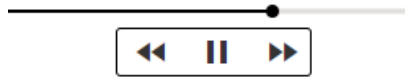
Backplane media streams

Media stream details are displayed when you have selected an individual node to view.

Field	Description
Type	Indicates whether the information is for an Audio , Video , or Presentation stream.
Start time	The time that the media stream started.
Tx codec	The format used by the transmitting Conferencing Node to encode and decode the media stream being transmitted.
Tx bitrate (kbps)	The quantity of data currently being sent from the transmitting Conferencing Node to the recipient Conferencing Node for this particular media stream.
Tx resolution	The display resolution of the image being sent from the transmitting Conferencing Node.
Tx framerate	The video frame rate per second being sent from the transmitting Conferencing Node.
Tx packets sent	The total quantity of packets sent from the transmitting Conferencing Node to the recipient Conferencing Node since the start of the conference.
Tx packets lost	The total quantity of packets sent from the transmitting Conferencing Node but not received by the recipient Conferencing Node.
Tx jitter (ms)	The variation in the expected periodic arrival of packets being sent from the transmitting Conferencing Node to the recipient Conferencing Node, in milliseconds.

Graph

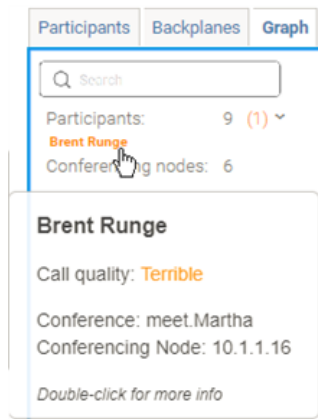
This section displays a dynamic graphical view of the connections for this conference, as follows:



You can use the timeline controls at the bottom of the graph to rewind and replay the graph at a variety of speeds. When viewing or replaying the graph you can:

- See when participants and Conferencing Nodes joined or disconnected from the conference.
- See when participants started and stopped presenting.
- View participant packet loss statistics during the conference by hovering over a connection.
- View summary details of individual participants, such as the protocol they are using and their bandwidth usage, by hovering over a participant. You can double-click on a participant to see more information.
- View summary details of individual nodes, such as its media load or any alarms, by hovering over a node. You can double-click on a node to see more information.
- Click within the graph to use your mouse to pan and zoom.

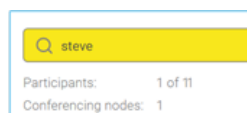
(See [Rewinding and replaying status](#) for more information about how to use the controls.)



Conference statistics and issues: the number of Conferencing Nodes and participants that are involved in the conference is displayed at the top left of the graph.

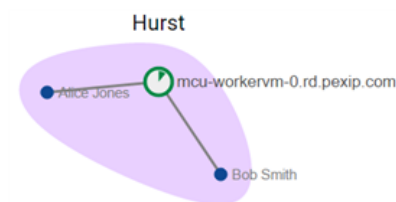
If any participants are experiencing call quality issues then the number of affected participants is displayed (in orange). You can click on this number to reveal the affected participants and also drill down to view more details about each of those participants.

The timeline indicates in blue any times when a participant or backplane had call quality issues. You can hover over these blue indicators to see more details of the issue.



Filtering: the Search box at the top left of the graph allows you to search for participants by name or alias.

When a filter is applied, any participants who match the filter text are highlighted in yellow. The timeline also indicates in yellow when there was a participant who matched the filter. You can hover over these yellow indicators to see more information about the match.






Colored areas: each colored area highlights a system location and shows the Conferencing Nodes and endpoint connections within that location. A different color is used for each location.

Marilou Enos
 208.205.178.128

 Role: Guest
 Protocol: WebRTC
 Bandwidth: 576kbps

Double-click for more info

Small dark blue dots: all participant endpoints. Some may have an icon next to their name, as follows:

-  for Infinity Connect presentation and control-only participants
-  for participants who are currently presenting content
-  for streaming participants.

You can hover over an endpoint to view participant information.



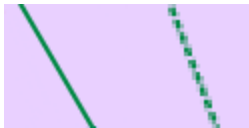
Large green circles: the Transcoding Conferencing Nodes to which the endpoints are connected, or are processing conference media. The amount of green fill within the circle indicates the current media load (in terms of percentage of estimated HD ports in use), so an unused node is white and a fully loaded node is filled entirely green.



Large blue circles: the Proxying Edge Nodes to which the endpoints are connected. The amount of green fill within the circle indicates the current media load (in terms of percentage of proxying capacity in use), so an unused node is white and a fully loaded node is filled entirely green.



Large pale blue circles: an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.



Green lines: backplane links between Conferencing Nodes, or links to external nodes. These become **dashed green lines** if total packet loss is greater than 1%.



Gray lines: connections between an endpoint and a Conferencing Node. These become **dashed gray lines** if total packet loss is greater than 1%.



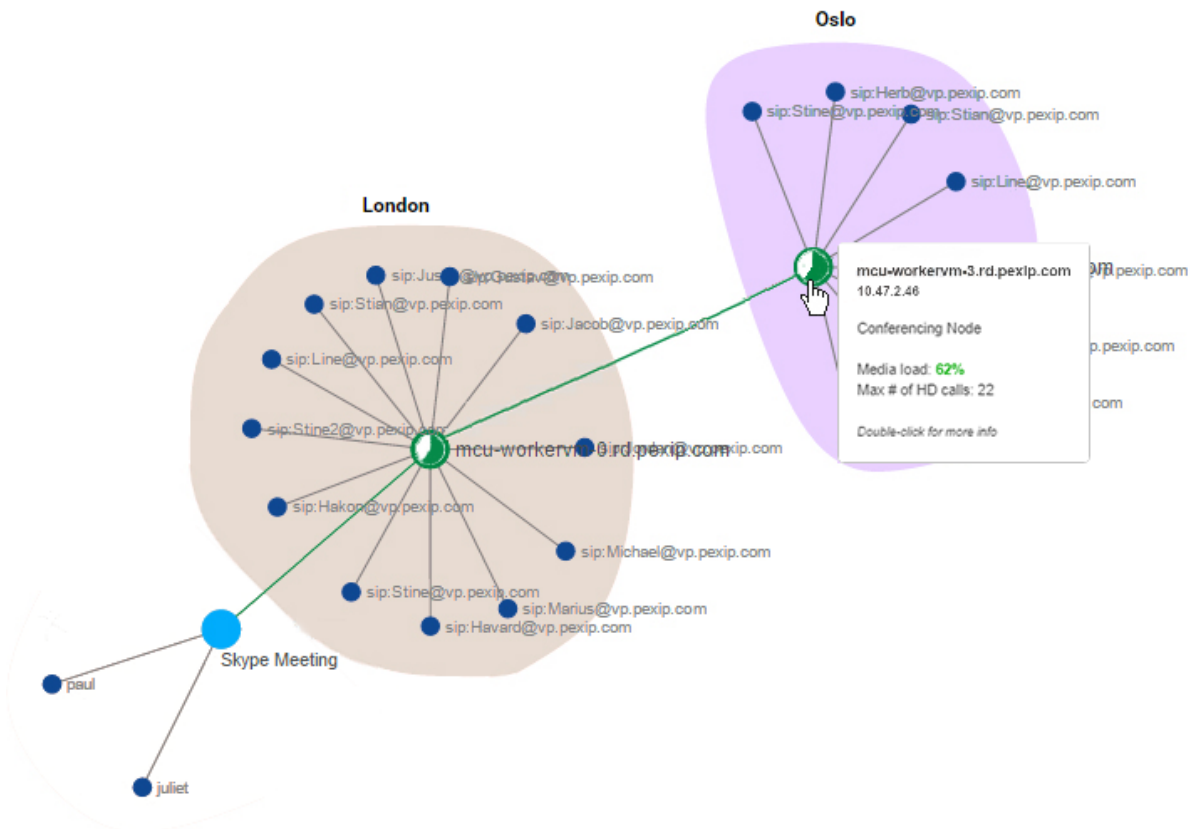
Red dashed lines: any connections with total packet loss greater than 2%.



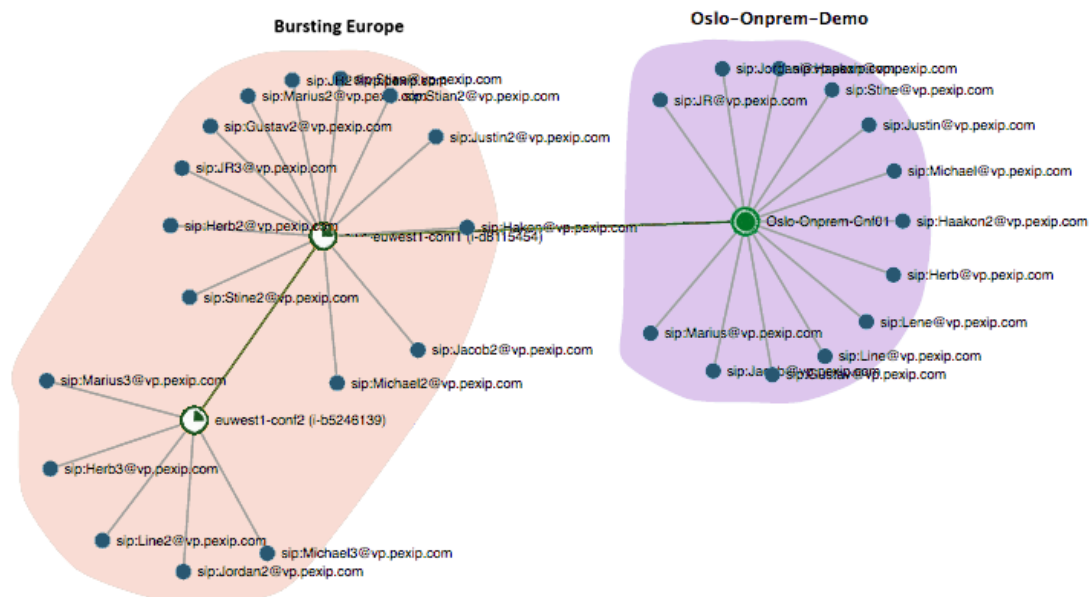
Blue lines: a media-forwarding link between a Proxying Edge Node and a Transcoding Conferencing Node. Only one link is shown regardless of how many connections/streams are being proxied. Packet loss information is not available on media-forwarding links.



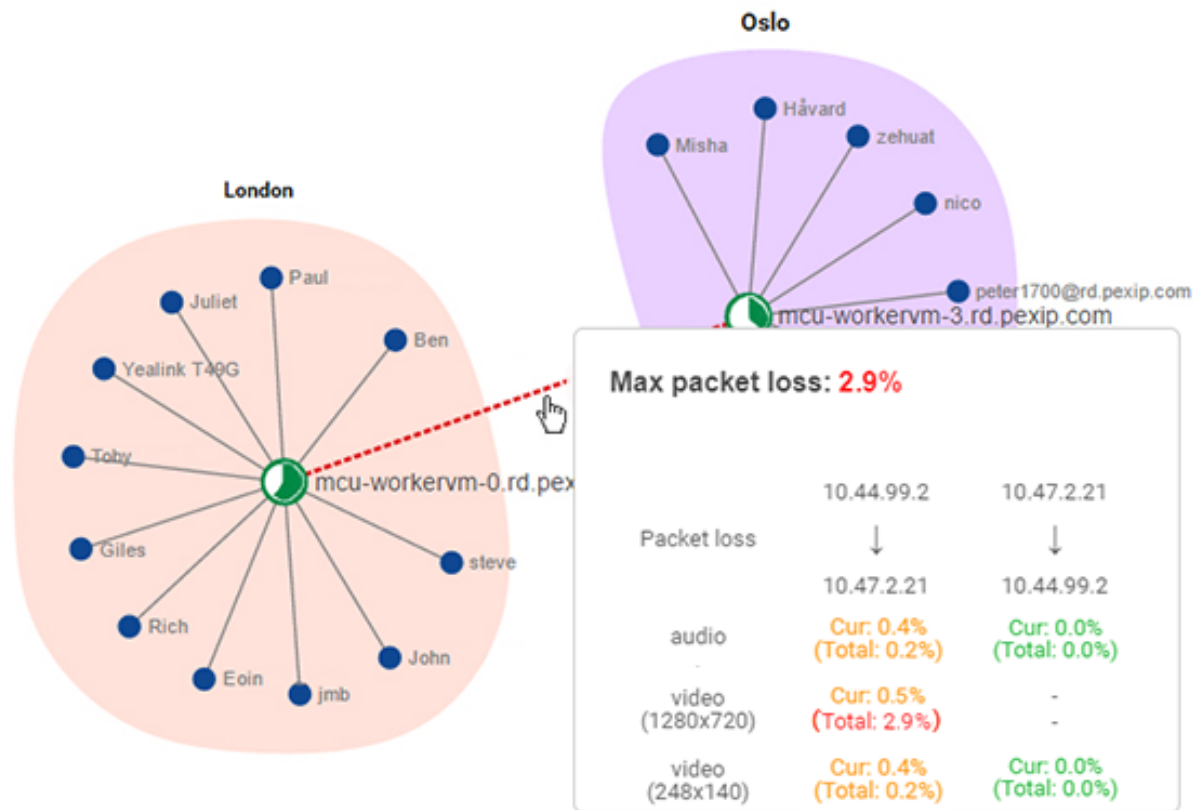
Keyhole: a keyhole in the top right of the screen indicates that the conference is locked.



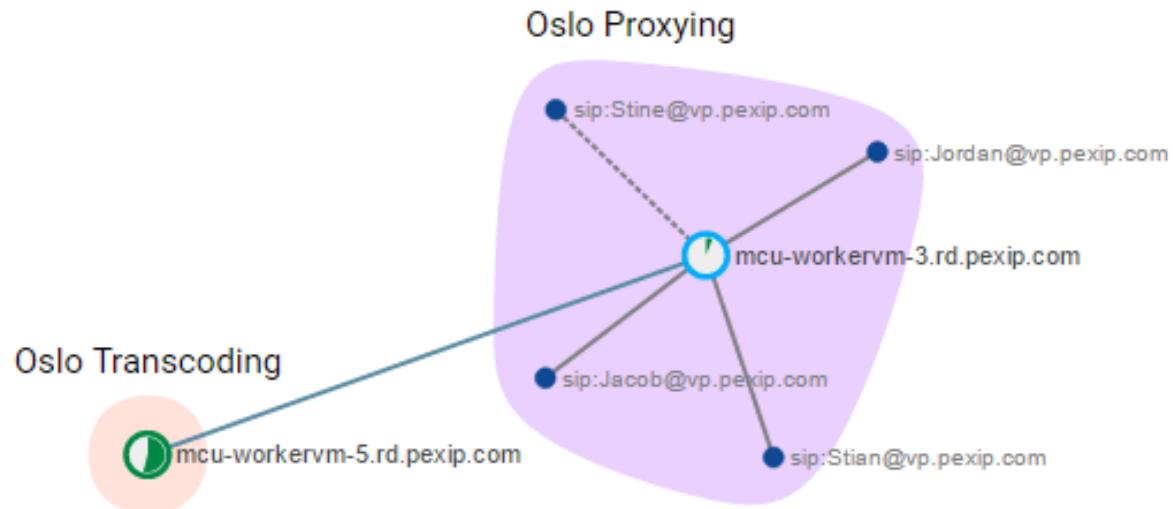
Example graph showing endpoints and a Skype for Business / Lync meeting connected to two Conferencing Nodes, where each Conferencing Node is in a separate location.



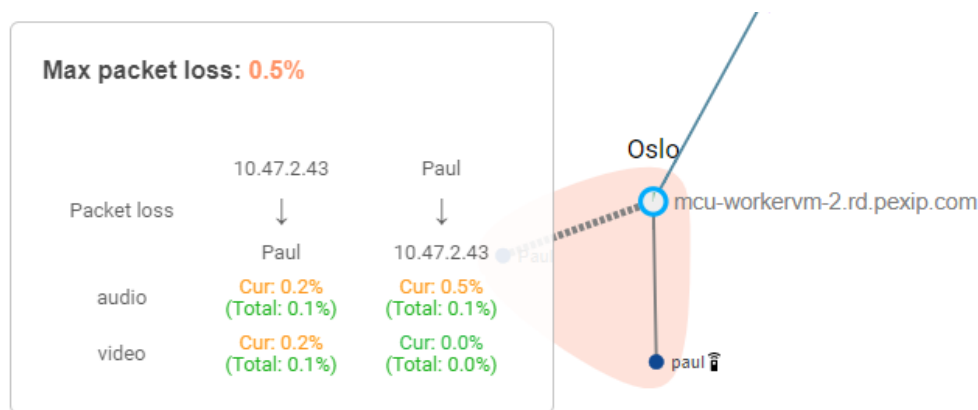
Example graph showing a system location containing two Conferencing Nodes that are hosted in AWS and are being used as an overflow location from an on-premises location containing one Conferencing Node.



Example graph showing packet loss on a backplane between two Conferencing Nodes in separate locations.



Example graph showing endpoints connected via a Proxying Edge Node to a VMR hosted on a Transcoding Conferencing Node in the "Oslo Transcoding" location.



Example graph showing packet loss for a connection.

Viewing participant status

To see a list of all the current conference participants across all Pexip Infinity services, go to **Status > Participants**. This shows a list of all participants; to view a particular participant's details, click on the **Participant alias**.

When viewing a participant's details you can also use the controls at the bottom of the page to:

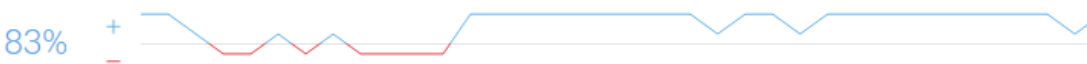
- **Mute** the participant
- **Disconnect** the participant from the conference
- **Transfer** the participant to another conference

Note that when viewing participants in a gateway call, these controls cannot be used on any remote participants that are connected to an externally-hosted conference.

To view historical information on participants after a conference has finished, see [Viewing historical information about participants](#).


Participant details

The following table lists the information shown for each participant. Note that some information, such as the Call ID, is not always available for participants who are directly connected to an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.

Field	Description
Perceived call quality *	<p>A graphical representation of the participant's call quality over time.</p> <p>A blue line at the top of the graph indicates Good, down to a red line at the bottom which indicates Terrible. The percentage number indicates the amount of the call where the quality is perceived as Good or OK (above the line in blue). For example:</p> <p>83% </p> <p>Note that a call quality of Unknown is reported for all calls of less than 20 seconds duration, and all calls over RTMP (of more than 20 seconds duration) always report a call quality of 100% Good, as they are placed over TCP.</p> <p>See media statistics and perceived call quality for more information.</p>
Participant alias	<p>The name of the user or the registered alias of the endpoint.</p> <p>Click on the participant alias to view detailed information about the call.</p>
Service name	<p>The name of the Virtual Meeting Room, Virtual Auditorium, Virtual Reception, Test Call Service or Call Routing Rule. For Infinity Gateway calls, the rule name is followed by a unique identifier to distinguish between separate calls.</p> <p>Select View status to view the current status of the conference.</p>

Field	Description
Call quality	The current quality of the call based on packet loss and jitter over the 3 most recent 20 second time windows. See media statistics and perceived call quality for more information.
Connect time *	The date and time that signaling was established between the participant's endpoint and Pexip Infinity.
Duration	The length of time since this participant joined the conference or accessed the service.
Display name	The name that has been configured on the participant's endpoint.
Destination alias *	For participants that have dialed in to the conference or service themselves, this is the alias that they dialed. For participants that have been dialed out to manually or automatically from a Virtual Meeting Room or Virtual Auditorium, this is the alias of the endpoint that was dialed.
System location	The system location of the Conferencing Node to which the endpoint is connected. However, when the participant is connected to a Proxying Edge Node, this is the location of the Transcoding Conferencing Node that is processing the conference media for this participant.
Proxying system location *	The system location of the Proxying Edge Node that is handling the call, if applicable.
Signaling node	The IP address and name of the Conferencing Node to which the endpoint is connected. This node is handling the call signaling but may or may not be handling the call media (for more information, see Handling of media and signaling in locally distributed conferences).
Media node	The IP address and name of the Transcoding Conferencing Node that is processing the call media for this participant (for more information, see Handling of media and signaling in locally distributed conferences).
Media proxying node *	The IP address and name of the Proxying Edge Node that is proxying the call media for this participant, if applicable.
Service type *	<p>Virtual Meeting Room, Virtual Auditorium, Virtual Reception, Test Call Service or Gateway: the participant has successfully accessed the service indicated.</p> <p>PIN collection IVR: the participant is currently accessing the Interactive Voice Response screens (where they are asked to enter a valid PIN).</p> <p>Waiting for Host: the participant is being shown a holding screen while they wait for a conference Host to join.</p> <p>Insufficient Capacity Screen: the participant is being shown a holding screen indicating that they cannot join the conference due to a lack of capacity on Pexip Infinity, or because the service's participant limit has been reached.</p> <p>Insufficient Licenses Screen: the participant is being shown a holding screen indicating that they cannot join the conference due to a lack of available call licenses on Pexip Infinity. For more information, see Insufficient licenses.</p> <p>Invalid License Screen: the participant is being shown a holding screen indicating that they cannot join the conference because there are no valid licenses available on Pexip Infinity. For more information, see Invalid license.</p>
Protocol	The communication protocol used by the endpoint.

Field	Description
Role	<p>Host indicates that either:</p> <ul style="list-style-type: none"> the conference has no PINs configured (in which case all participants have a role of Host) the participant accessed the conference using the Host PIN. <p>Guest indicates that the participant accessed the conference using the Guest PIN.</p> <p>Unknown indicates one of the following:</p> <ul style="list-style-type: none"> the participant is at the PIN entry screen and has not yet successfully entered a PIN the participant is at the Waiting for Host screen but their role has not yet been determined the participant is connected to an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet (these participants appear in the Infinity Connect participant list with a role of External Guest). <p>For more information, see About PINs, Hosts and Guests.</p>
License count *	<p>The number of licenses consumed by this participant. Media participants consume 1 license but API-only participants (e.g. Infinity Connect users who are not sending media) do not consume a license.</p> <p>Participants who are directly connected to an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet, do not consume a license.</p>
License type *	<p>The type of license used, either:</p> <ul style="list-style-type: none"> Port: audio/video participant. Audio: audio-only participant — only applies if the system has audio licenses installed, otherwise port licenses are used for audio-only calls) Not required: a presentation and control-only participant, or a participant directly connected to an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.
Is presentation stream supported *	Indicates whether the endpoint is able to support a separate media stream for presentations negotiated by H.239 or BFCP.
Is presenting	Indicates whether the endpoint is currently sending a presentation stream.
Is muted	Indicates whether the endpoint's audio has been muted (using an Infinity Connect client, the Administrator interface, or by a third party using the Pexip API).
On hold *	Indicates whether the endpoint has been put on hold (usually by the endpoint user).
Call direction *	<p>In: the call was placed by an external endpoint and received by Pexip Infinity.</p> <p>Out: the call was placed by Pexip Infinity to an endpoint or other device.</p>
Bandwidth (kbps) *	The maximum bandwidth, in kbps, negotiated for use between the Conferencing Node and the endpoint. Actual bandwidth used is shown in the Media streams section (Tx bitrate and Rx bitrate).
Streaming or recording device *	Indicates if the participant is a streaming or recording device.
Encryption *	Indicates whether the media stream being sent to and from the Conferencing Node towards the endpoint is encrypted.
Vendor *	Information about the endpoint's manufacturer and software.
Remote IP address *	The IP address of the system from which signaling from this endpoint is being sent and received. This may be the endpoint itself, or it may be a call control system if one is in use in your network.
Remote port *	The port on the system from which signaling from this endpoint is being sent and received.

Field	Description
Call ID *	<p>A unique identifier that can be used to trace the call in the administrator log and support log.</p> <p>Select View call logs to see a filtered view of the support log showing only events containing this Call ID.</p> <p>Select View log summary to see a condensed view of the call signaling messages in the support log for this Call ID.</p> <p> Calls made via the Virtual Reception generate two separate participant calls but these both have the same Call ID. Calls made via the Infinity Gateway generate separate participant calls with different Call IDs.</p>
Is disconnect / transfer / mute supported *	Indicates whether the participant can be disconnected, transferred or muted via the Management Node.
* Only displayed when you have selected an individual participant to view.	

Media streams

Media stream details are displayed when you view the details of an individual participant. Note that media streams are not displayed for any participants who are directly connected to an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.

Type	<p>Indicates whether the information is for the Audio, Video, or Presentation stream.</p> <p>When viewing the stream details of a completed VMR call you may see multiple instances of each stream type (for example, if the participant had started presenting, stopped and then started presenting again). For in-progress VMR calls you only see a maximum of one instance of each stream type (reflecting what the participant is currently doing).</p> <p>In gateway calls to an externally-hosted conference you may see a separate stream for every resolution/frame rate being sent.</p> <p>Note that a presentation stream is not shown for Infinity Connect clients that are sending or receiving still images or PDF pages (as opposed to screen sharing, or receiving full motion presentation).</p>
Start time	The date and time that the media stream started.
Node	The address of the Transcoding Conferencing Node handling the media.
Tx codec	The format used by the Conferencing Node to encode the media stream being sent to the endpoint.
Tx bitrate (kbps)	The quantity of data currently being sent from the Conferencing Node to the endpoint, in kilobits per second.
Tx resolution	The display resolution of the image being sent from the Conferencing Node.
Tx framerate	The video frame rate per second being sent from the Conferencing Node.
Tx packets sent	The total quantity of packets sent from the Conferencing Node to the endpoint since the start of the conference.
Tx packets lost	<p>The total quantity of packets sent from the Conferencing Node but not received by the endpoint.</p> <p>This value is reported to the Conferencing Node by the endpoint. Endpoints that do not support RTCP are not able to supply this information, so the value will always be 0.</p>
Tx jitter (ms)	<p>The variation in the expected periodic arrival of packets being sent from the Conferencing Node to the endpoint, in milliseconds.</p> <p>This value is reported to the Conferencing Node by the endpoint. Endpoints that do not support RTCP cannot supply this information, so the value will always be 0.</p>

Rx codec	<p>The format used by the Conferencing Node to decode the media stream being sent from the endpoint.</p> <p><i>Off</i> indicates that no decodable media for this stream has been received in the last 10 seconds or so from the codec. For example, for an Infinity Connect client being used for presentation and control-only, there will be no video transmitted.</p> <p><i>Off stage</i> indicates that the participant is currently not being shown in the main video or thumbnails, so Pexip Infinity is not attempting to decode the media stream.</p> <p><i>Telephone event</i> may be displayed if an audio codec that use silence suppression (such as G729B) is muted and sends DTMF.</p> <p>If this field is blank, this may mean that Pexip Infinity has negotiated which codec to use but it is yet to receive any media from the endpoint to determine which codec it is actually sending.</p>
Rx bitrate (kbps)	The quantity of data currently being received by the Conferencing Node from the endpoint, in kilobits per second.
Rx resolution	The display resolution of the image being received by the Conferencing Node.
Rx framerate	The video frame rate per second being received by the Conferencing Node. This can fluctuate over time as it is measured by the Conferencing Node.
Rx packets received	The total quantity of packets received by the Conferencing Node from the endpoint since the start of the conference.
Rx packets lost	The total quantity of packets sent from the endpoint but not received by the Conferencing Node.
Rx jitter (ms)	The variation in the expected periodic arrival of packets being received by the Conferencing Node from the endpoint, in milliseconds.

Viewing registrations

To see a list of all device aliases that are currently registered to the Pexip Infinity platform, go to **Status > Registrations**.

Devices can only register to Pexip Infinity if the alias it wants to register has been added to Pexip Infinity's list of allowed device aliases (**Users & Devices > Device Aliases**). For more information, see [Registering devices to Pexip Infinity](#).

The following information about each registration is available:

Field	Description
Alias	<p>The alias that has been registered, including its URI scheme (e.g. sip:).</p> <p>Click on the alias to view detailed status information.</p>
Username	The username associated with the alias that was used to authenticate the registration.
Node	The address of the node to which the device is registered.
Start time	The date and time the registration started.
Protocol	The protocol over which the device alias is registered, such as SIP, H.323 or WebRTC.
Is natted *	Indicates if the registered device is probably behind a NAT (its contact address is different from its source IP address).
* Only displayed when you have selected an individual alias to view.	

Viewing historical information about conferences

To see a list of all the completed conferences on the Pexip Infinity platform, go to **History & Logs > Conference History**. This shows a list of the most recent completed conferences (up to a limit of 10,000). To view details of a particular conference, including the ability to rewind and replay the conference [graph](#), click on the **Service name**.

To view information on conferences currently running on Pexip Infinity, see [Viewing current conference status](#).

The Administrator interface uses color coding when reporting media statistics, such as perceived call quality, packet loss and jitter. In general, statistics that are shown in green represent good quality, orange represents intermediate quality, and red is used for bad quality. See [media statistics and perceived call quality](#) for more information.

The following information is available for each completed conference:

Field	Description
Service name	<p>The name of the Virtual Meeting Room, Virtual Auditorium, Virtual Reception, Test Call Service or Call Routing Rule. For Infinity Gateway calls, the rule name is followed by a unique identifier to distinguish between separate calls.</p> <p>For Virtual Receptions and Test Call Services, if there were multiple concurrent users of that service you will see a single instance of that service (rather than one instance per participant, as all participants are using the same service even though they cannot see or hear each other).</p> <p>Click on the service name to view more information.</p>
Start time	The date and time that the first participant connected to the service.
End time	The date and time that the last participant's call ended.
Duration	The length of time that the conference or service was in use.
Participant count	The total number of participant calls made to this conference. Note that if a single participant disconnects from the conference and then reconnects to it, this will be counted as two participant calls.
Service type *	The type of conference, e.g. Virtual Meeting Room.
Service tag *	The unique identifier that an administrator has assigned to this service. If this field is blank, no tag has been assigned. For more information, see Tracking usage via service and participant call tags .
Instant message count *	The total number of instant messages sent during the conference.
* Only displayed when you have selected an individual conference to view.	

To view more information about the conference, click on the service name. You will then see 3 tabs for the selected conference: [Participants](#), [Backplanes](#) and [Graph](#).

Participants

The **Participants** section lists all the participants that were in the conference.

For more details about a particular participant, including media stream statistics, click on the **Participant alias**. This takes you to the [Participant history](#) page.

Field	Description
Participant alias	The name of the user or the registered alias of the endpoint.
Start time	The date and time that the participant's call reached Pexip Infinity.
End time	The date and time that the participant's call ended.
Duration	The length of time that the participant was connected to Pexip Infinity. This includes time connected to the Virtual Meeting Room or Virtual Auditorium, and any time spent at the Virtual Reception or PIN entry screens.
Display name	The name that has been configured on the participant's endpoint.
System location	The system location of the Conferencing Node to which the endpoint is connected. However, when the participant is connected to a Proxying Edge Node, this is the location of the Transcoding Conferencing Node that is processing the conference media for this participant.

Field	Description
Role	<p>Host indicates that either:</p> <ul style="list-style-type: none"> the conference had no PINs configured (in which case all participants had a role of Host) the participant accessed the conference using the Host PIN. <p>Guest indicates that the participant accessed the conference using the Guest PIN.</p> <p>Unknown indicates one of the following:</p> <ul style="list-style-type: none"> the participant reached the Virtual Reception but did not proceed to a Virtual Meeting Room or Virtual Auditorium the participant reached the PIN entry screen but did not successfully enter a PIN the participant reached the Waiting for Host screen but their role was not determined the call was via the Infinity Gateway. <p>For more information, see About PINs, Hosts and Guests.</p>

Backplanes

The **Backplanes** section provides information about the media streams being transmitted between Transcoding Conferencing Nodes for the selected conference. Backplane links between Conferencing Nodes are unidirectional, so for a conference involving two transcoding nodes there will be two backplane links: one from node A to node B, and another from node B to node A. Note that a bidirectional backplane is created when a Conferencing Node connects to a Teams Connector or to a Skype for Business / Lync meeting.

Field	Description
Media node	<p>The IP address and name of the Conferencing Node that is transmitting media.</p> <p>For details about the media streams being sent over a particular backplane link, click on the media node's IP address.</p>
Remote media node	<p>The IP address and name of the Conferencing Node or remote system e.g. a Teams Connector, that is receiving media.</p> <p>Note that the remote media node of a merged SfB/Lync meeting is identified by the address of the SfB/Lync client that initiated the SfB/Lync meeting.</p>
Remote conference name *	The name of the conference on the remote node. For external backplanes, this identifies the conference on the other platform, such as a Microsoft Teams conference ID.
System location *	The system location of the Conferencing Node that is transmitting media.
Start time	The date and time that the connection was established.
End time	The date and time that the connection was brought down.
Duration	The length of time since the connection was established.
Backplane type	<p>Geographic indicates that the two Conferencing Nodes are in different system locations.</p> <p>Local indicates that the two Conferencing Nodes are in the same system location.</p> <p>External indicates a link between a Conferencing Node and an external node, such as a Teams Connector.</p>
Disconnect reason *	The reason that the backplane link was disconnected.
* Only displayed when you have selected an individual media node to view.	

Backplane media streams

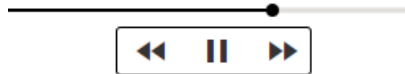
Media stream details are displayed when you have selected an individual node to view.

Field	Description
Type	Indicates whether the information is for an Audio , Video , or Presentation stream.
Start time	The time that the media stream started.
End time	The time that the media stream ended.
Tx codec	The format used by the transmitting Conferencing Node to encode and decode the media stream being transmitted.
Tx bitrate (kbps)	The quantity of data currently being sent from the transmitting Conferencing Node to the recipient Conferencing Node for this particular media stream.
Tx resolution	The display resolution of the image being sent from the transmitting Conferencing Node.
Tx framerate	The video frame rate per second being sent from the transmitting Conferencing Node.
Tx packets sent	The total quantity of packets sent from the transmitting Conferencing Node to the recipient Conferencing Node since the start of the conference.
Tx packets lost	The total quantity of packets sent from the transmitting Conferencing Node but not received by the recipient Conferencing Node.

Graph

This section displays a dynamic graphical view of the connections for this conference, as described below.

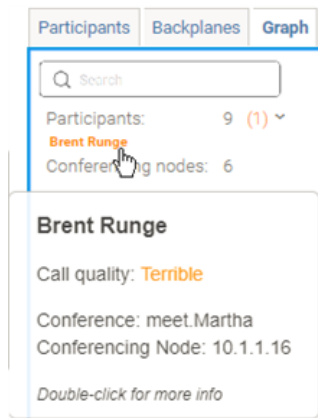
- i** Initially the graph shows the midpoint state of the conference. You can use the interactive timeline and controls to go forwards or backwards to replay the graph and review the entire conference.



You can use the timeline controls at the bottom of the graph to rewind and replay the graph at a variety of speeds. When viewing or replaying the graph you can:

- See when participants and Conferencing Nodes joined or disconnected from the conference.
- See when participants started and stopped presenting.
- View participant packet loss statistics during the conference by hovering over a connection.
- View summary details of individual participants, such as the protocol they are using and their bandwidth usage, by hovering over a participant. You can double-click on a participant to see more information.
- View summary details of individual nodes, such as its media load or any alarms, by hovering over a node. You can double-click on a node to see more information.
- Click within the graph to use your mouse to pan and zoom.

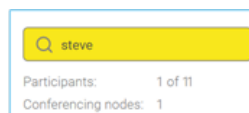
(See [Rewinding and replaying status](#) for more information about how to use the controls.)



Conference statistics and issues: the number of Conferencing Nodes and participants that are involved in the conference is displayed at the top left of the graph.

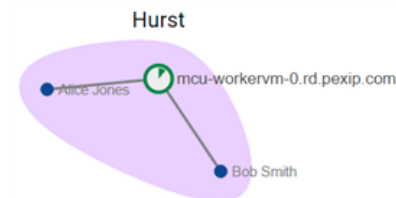
If any participants are experiencing call quality issues then the number of affected participants is displayed (in orange). You can click on this number to reveal the affected participants and also drill down to view more details about each of those participants.

The timeline indicates in blue any times when a participant or backplane had call quality issues. You can hover over these blue indicators to see more details of the issue.

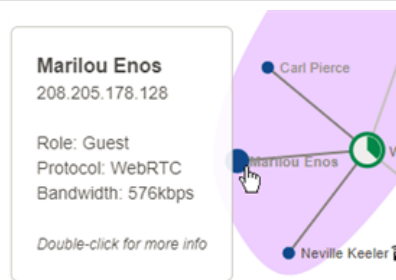


Filtering: the Search box at the top left of the graph allows you to search for participants by name or alias.




When a filter is applied, any participants who match the filter text are highlighted in yellow. The timeline also indicates in yellow when there was a participant who matched the filter. You can hover over these yellow indicators to see more information about the match.



Colored areas: each colored area highlights a system location and shows the Conferencing Nodes and endpoint connections within that location. A different color is used for each location.



Small dark blue dots: all participant endpoints. Some may have an icon next to their name, as follows:

-  for Infinity Connect presentation and control-only participants
-  for participants who are currently presenting content
-  for streaming participants.

You can hover over an endpoint to view participant information.



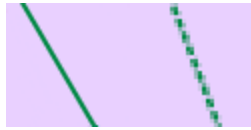
Large green circles: the Transcoding Conferencing Nodes to which the endpoints are connected, or are processing conference media. The amount of green fill within the circle indicates the current media load (in terms of percentage of estimated HD ports in use), so an unused node is white and a fully loaded node is filled entirely green.



Large blue circles: the Proxying Edge Nodes to which the endpoints are connected. The amount of green fill within the circle indicates the current media load (in terms of percentage of proxying capacity in use), so an unused node is white and a fully loaded node is filled entirely green.



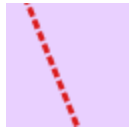
Large pale blue circles: an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.



Green lines: backplane links between Conferencing Nodes, or links to external nodes. These become **dashed green lines** if total packet loss is greater than 1%.



Gray lines: connections between an endpoint and a Conferencing Node. These become **dashed gray lines** if total packet loss is greater than 1%.



Red dashed lines: any connections with total packet loss greater than 2%.



Blue lines: a media-forwarding link between a Proxying Edge Node and a Transcoding Conferencing Node. Only one link is shown regardless of how many connections/streams are being proxied. Packet loss information is not available on media-forwarding links.



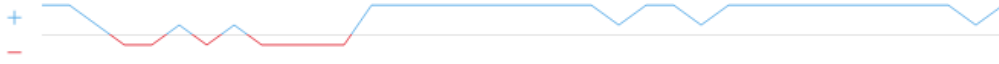
Keyhole: a keyhole in the top right of the screen indicates that the conference is locked.

Viewing historical information about participants


To see a list of all the calls made to the Pexip Infinity platform, go to **History & Logs > Participant History**. This shows a list of all calls made to the 5,000 most recent completed conferences. To view details of a particular call, click on the **Participant alias**.

To view information on participants currently connected to Pexip Infinity, see [Viewing participant status](#).

The following table lists the information shown for each participant. Note that some information, such as the Call ID, is not always available for participants who are directly connected to an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.

Field	Description
Perceived call quality *	<p>A graphical representation of the participant's call quality over time.</p> <p>A blue line at the top of the graph indicates Good, down to a red line at the bottom which indicates Terrible. The percentage number indicates the amount of the call where the quality is perceived as Good or OK (above the line in blue). For example:</p> <p>83% </p> <p>Note that a call quality of Unknown is reported for all calls of less than 20 seconds duration, and all calls over RTMP (of more than 20 seconds duration) always report a call quality of 100% Good, as they are placed over TCP.</p> <p>See media statistics and perceived call quality for more information.</p>

Field	Description
Participant alias	<p>The name of the user or the registered alias of the endpoint.</p> <p>Click on the participant alias to view detailed information about the call.</p>
Service name	<p>The name of the Virtual Meeting Room, Virtual Auditorium, Virtual Reception, Test Call Service or Call Routing Rule. For Infinity Gateway calls, the rule name is followed by a unique identifier to distinguish between separate calls.</p> <p>Select View conference to view the historical status of the conference, and optionally to replay the conference graph.</p>
Call quality	<p>The perceived overall quality of the call.</p> <p>See media statistics and perceived call quality for more information.</p>
Start time	The date and time that the participant's call reached Pexip Infinity.
End time	The date and time that the participant's call ended.
Duration	The length of time that the participant was connected to Pexip Infinity. This includes time connected to the Virtual Meeting Room or Virtual Auditorium and any time spent at the Virtual Reception or PIN entry screens.
Display name	The name that has been configured on the participant's endpoint.
Conference alias *	The alias that the participant dialed to access the service. If the participant included a PIN number in the dial string, this would also be included in the alias shown here.
System location	The system location of the Conferencing Node to which the endpoint is connected. However, when the participant is connected to a Proxying Edge Node, this is the location of the Transcoding Conferencing Node that is processing the conference media for this participant.
Proxying system location *	The system location of the Proxying Edge Node that is handling the call, if applicable.
Signaling node *	The IP address and name of the Conferencing Node to which the endpoint is connected. This node is handling the call signaling but may or may not be handling the call media (for more information, see Handling of media and signaling in locally distributed conferences).
Media node	The IP address and name of the Transcoding Conferencing Node that is processing the call media for this participant (for more information, see Handling of media and signaling in locally distributed conferences).
Media proxying node *	The IP address and name of the Proxying Edge Node that is proxying the call media for this participant, if applicable.
Service type *	Indicates whether the participant was connected to a Virtual Meeting Room, Virtual Auditorium, Virtual Reception, Test Call Service or the Infinity Gateway.
Protocol	The communication protocol used by the endpoint.

Field	Description
Role	<p>Host indicates that either:</p> <ul style="list-style-type: none"> the conference had no PINs configured (in which case all participants had a role of Host) the participant accessed the conference using the Host PIN. <p>Guest indicates that the participant accessed the conference using the Guest PIN.</p> <p>Unknown indicates one of the following:</p> <ul style="list-style-type: none"> the participant reached the Virtual Reception but did not proceed to a Virtual Meeting Room or Virtual Auditorium the participant reached the PIN entry screen but did not successfully enter a PIN the participant reached the Waiting for Host screen but their role was not determined the call was via the Infinity Gateway. <p>For more information, see About PINs, Hosts and Guests.</p>
License count *	<p>The number of licenses consumed by this participant. Media participants consume 1 license but API-only participants (e.g. Infinity Connect users who are not sending media) do not consume a license.</p> <p>Participants who are directly connected to an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet, do not consume a license.</p>
License type *	<p>The type of license used, either:</p> <ul style="list-style-type: none"> Port: audio/video participant. Audio: audio-only participant — only applies if the system has audio licenses installed, otherwise port licenses are used for audio-only calls) Not required: a presentation and control-only participant, or a participant directly connected to an externally-hosted conference, such as a Microsoft Teams or Skype for Business meeting, or Google Meet.
Call direction *	<p>In: the call was placed by an external endpoint and received by Pexip Infinity.</p> <p>Out: the call was placed by Pexip Infinity to an endpoint or other device.</p>
Bandwidth (kbps) *	<p>The maximum bandwidth, in kbps, negotiated for use between the Conferencing Node and the endpoint. Actual bandwidth used is shown in the Media streams section (Tx bitrate and Rx bitrate).</p>
Streaming or recording device *	<p>Indicates if the participant is a streaming or recording device.</p>
Encryption *	<p>Indicates whether the media stream being sent to and from the Conferencing Node towards the endpoint is encrypted.</p>
Vendor *	<p>Information about the endpoint's manufacturer and software.</p>
Remote IP address *	<p>The IP address of the system from which signaling from this endpoint is being sent and received. This may be the endpoint itself, or it may be a call control system if one is in use in your network.</p>
Remote port *	<p>The port on the system from which signaling from this endpoint is being sent and received.</p>
Call ID *	<p>A unique identifier that can be used to trace the call in the administrator log and support log.</p> <p>Select View call logs to see a filtered view of the support log showing only events containing this Call ID.</p> <p>Select View log summary to see a condensed view of the call signaling messages in the support log for this Call ID.</p> <p> Calls made via the Virtual Reception generate two separate participant calls but these both have the same Call ID. Calls made via the Infinity Gateway generate separate participant calls with different Call IDs.</p>
Disconnect reason	<p>The reason that the call was disconnected. This is provided by either the endpoint or Pexip Infinity, depending on how the call was terminated. For more information, see Disconnection reasons.</p>
* Only displayed when you have selected an individual participant to view.	

Media streams

Historical [media stream details](#) are displayed when you view the details of an individual participant.

Reporting of media statistics and perceived call quality

When using the Administrator interface to look at the current and historic status of calls and of individual participants within a call, various media statistics and guidance as to perceived call quality are displayed.

Perceived call quality

The current and historic participant status screens provide guidance to the perceived quality of a call by looking at packet loss and jitter over multiple time windows.

For an ongoing call, the system looks at packet loss and jitter over the 3 most recent 20 second time windows. To calculate call quality for a time window, packet loss is assessed first: < 1% packet loss is perceived as Good quality; < 3% is OK; < 10% is Bad; otherwise it is Terrible. Then, if jitter is > 40ms the call quality assessment based on packet loss is adjusted to the next level down (e.g. OK is reduced to Bad).

The system then reports the current call quality as the most frequent call quality in those last 3 windows. If all time windows report a different quality, then any of the three qualities may be indicated.

For completed calls, the system looks at packet loss and jitter over multiple 20 second time windows throughout the call and calculates the call quality per window. It then reports the overall call quality as the average call quality of those windows.

The system also provides a graphical representation of the participant's call quality over the duration of the call.

A blue line at the top of the graph indicates Good, down to a red line at the bottom which indicates Terrible. The percentage number indicates the amount of the call where the quality is perceived as Good or OK (above the line in blue). For example:



Note that a call quality of Unknown is reported for all calls of less than 20 seconds duration, and all calls over RTMP (of more than 20 seconds duration) always report a call quality of 100% Good, as they are placed over TCP.

Color coding for media statistics and quality

The Administrator interface uses color coding when reporting media statistics, such as perceived call quality, packet loss and jitter. In general, statistics that are shown in **green** text represent good quality, **orange** represents intermediate quality, and **red** is used for bad quality.

- **Perceived call quality** is calculated as described above and is shown in **green** if the quality is Good or OK, or in **red** if the quality is Bad or Terrible.
- **Packet loss** is shown in **green** when it is < 0.2%, in **orange** when it is between 0.2% and 2%, and in **red** when it is >= 2%.
- **Jitter** is shown in **green** when it is < 10ms, in **orange** when it is 10–50ms, and in **red** when it is >= 50ms.

Viewing usage statistics

To view the daily usage statistics per system location, go to **History & Logs > Usage Statistics**. This shows:

- the total number of minutes used (the sum of all participant durations), and
- the highest number of concurrent call licenses used.

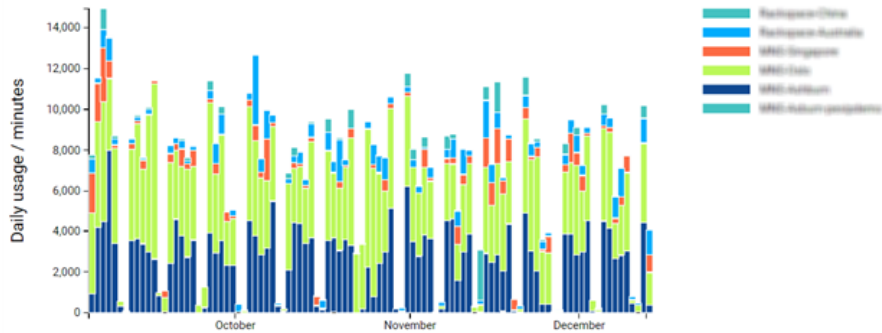
These graphs show statistics for up to the past 100 days, even if you have rebooted or upgraded your system during that time. Each day represents usage for the 24-hour period from midnight UTC. The information is also broken down by ingress location within each day.

You can view specific information on each location and day by hovering over each item.

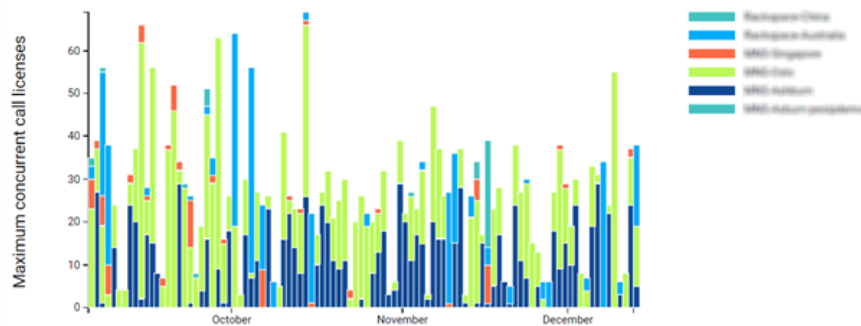
If there is no information available, for example if your system has been deployed in the last 24 hours, you will see the following message: **Information not yet available. This can take up to 1 hour.**

- i** The [Live view](#) page (**Status > Live View**) lets you review current and historic usage charts showing a breakdown of participants by location, protocol, license type and the different conference types being hosted.

Example graphs



Graph showing usage in participant minutes per day for a deployment spread across six locations



Graph showing the maximum number of concurrent call licenses used each day for a deployment spread across six locations

Viewing source data

To view the source data used to create each graph, click on the [View source data](#) link below the graph. This takes you to a page showing the source data in CSV format.

The source data goes back for the same amount of time as the administrator log, from which they are extracted. For more information, see [About the administrator log](#).

Viewing LDAP sync template results

To view the status of ongoing or completed [LDAP template synchronization](#) processes, go to **Status > LDAP Sync**.

You are shown the details of the most recent or in-progress run of each LDAP synchronization template. Information shown includes:

- run status e.g. Initializing sync: connecting to LDAP, Syncing, Sync succeeded
- number of VMRs, devices and users that were created, deleted, updated and unchanged
- number of warnings
- last updated date and time.

If necessary, you can select a template to view the details of the last error/warning message it generated. Typical messages include:

Warning	Comment
Error syncing with LDAP	This can occur if Pexip Infinity fails to connect to, or authenticate with, the LDAP server. In which case the Pexip Infinity support log will provide more information. Alternatively, this can be caused by invalid syntax in the template's LDAP user filter or LDAP user search DN fields.
Alias clash: alias held by an existing conference has not been assigned to the newly synced conference	This occurs when a template generates an alias that is identical to an existing alias that is already assigned to another VMR (either created manually or via another mechanism). In this case, the alias is not reassigned to the VMR currently being created/updated.
Duplicate alias: attempting to create the same alias more than once for the same conference	This occurs when the template attempts to create the same alias more than once for the same conference. Check the alias patterns for your template.
Conference clash: not overwriting manually created conference	This occurs when a template generates a VMR name that already exists, and that VMR was created manually. In this case, the existing VMR with that name is left unchanged.
Sync template clash: not overwriting conference created by another mechanism	This occurs when a template generates a VMR name that already exists, and that VMR was created via another mechanism such as a different template or VMR Scheduling for Exchange. In this case, the existing VMR with that name is left unchanged.
Validation error when creating conference	This occurs if the synchronization tries to set a field to an invalid value (e.g. a PIN containing letters rather than numbers, or an empty VMR name). The error message will include some addition parameters such as Conference, Error and Data which will help to identify the VMR, the field and source LDAP data that is causing the problem.
Validation error when creating device	This occurs if the synchronization tries to set a field to an invalid value (such as an empty alias). The error message will include some addition parameters such as Device-Alias, Error and Data which will help to identify the device alias, field and source LDAP data that is causing the problem.
Sync template clash: not overwriting device created by another template	This occurs when a template generates a device alias that already exists, and that alias was created via another template. In this case, the existing device alias is left unchanged.
Device clash: not overwriting manually created device	This occurs when a template generates a device alias that already exists, and that alias was created manually. In this case, the existing device alias is left unchanged.
Sync template clash: not overwriting User created by another template	This occurs when a template generates a user email address that already exists, and that user record was created via another template. In this case, the existing user record is left unchanged.
User clash: not overwriting manually created User	This occurs when a template generates a user email address that already exists, and that user record was created manually. In this case, the existing user record is left unchanged.

All of the warning messages generated from a sync run can be viewed in the Administrator log (**Status > Administrator Log**) as described below.

Viewing all historical sync results via the Administrator log

To view all historic details of completed LDAP synchronization processes:

1. Go to the Administrator log (**Status > Administrator Log**).
2. Search for **ldap.sync** to see details of all log entries relating to all VMR, device and user synchronization processes.

Note that you will see "Beginning" and "Completed" log entries for VMR, device and user syncing even if the template is not configured to perform both types of syncing. This is because all sync processes are always performed in case they need to delete VMRs, devices or users from previous synchronizations.

More information

For more information about bulk-provisioning VMRs and devices from an LDAP-accessible database, see:

- [Provisioning VMRs, devices and users from Active Directory via LDAP](#)
- [Troubleshooting LDAP server connections](#)

Diagnostics tools and reporting

Automatically reporting errors

In order to allow continual monitoring and improvement of the Pexip Infinity platform, we encourage customers to send us details of any incidents that occur in their products. When this feature is enabled, incident reports are sent automatically to a specified URL. By default this is the address of a secure web server owned and managed by Pexip, but you have the option to change the URL.

We collate and analyze this information in order to identify issues with the software that can be resolved in future product releases. The information we receive will not be used for any other purpose.

The feature is [enabled and disabled](#) on a Pexip Infinity platform-wide basis, but when enabled each Management Node and Conferencing Node will send their individual incident reports directly to the incident reporting server.

When incident reporting is enabled:

- If you are [Using a web proxy](#) for the Management Node, or for the Location to which the Conferencing Node belongs, all incident reports will be sent via the specified web proxy.
- If you are not using a web proxy, you must ensure that your IP routing and firewall configuration allows outbound communications from each of these systems to the nominated URL. By default, reports are sent to <https://acr.pexip.com> over TLS to port 443.

Content of incident reports

The exact content of each incident report will vary depending on the nature of the problem, but all reports **always** include:

- the date and time that the incident occurred
- the FQDN of the system on which the incident occurred
- the IP address of the system on which the incident occurred
- information about the Pexip Infinity software version
- details of the internal software processes running at the time the incident occurred.

The incident reporting server will also annotate each report with the IP address from which it was received (generally this will be the public IP address of the sender's network). This information is not contained in the original incident report.

The incident reports **may** also include:

- system configuration information
- stack traces.

The incident reports **never** include:

- system logs.

Enabling and disabling automatic sending of incident reports

Automatic sending of incident reports is enabled or disabled during initial installation of the Management Node (when running the installation wizard).

To enable or disable the automatic submission of incident reports, or to change the URL to which they are sent:

1. On the Pexip Infinity Administrator interface, go to **Platform > Global Settings**.
2. From the **Reporting** section, select or deselect the **Enable incident reporting** box as required.
3. If appropriate, in the **Incident reporting URL** field enter the URL to which reports will be sent.
4. Optionally, enter a **Contact email address** of somebody who can be contacted by Pexip for further information. If specified, this email address is added to incident reports.
5. Select **Save**.

Tracking usage via service and participant call tags

To assist in tracking usage of the Pexip Infinity platform you can assign **service tags** to conferencing services such as VMRs, Virtual Auditoriums, Virtual Receptions and Call Routing Rules, and **participant call tags** to each participant within a conference.

You can also use the service and participant call tags for decision-making purposes in local and external policy.

Service tags

Each Virtual Meeting Room (including those used for scheduled conferences), Virtual Auditorium, Virtual Reception, Call Routing Rule and registered device can be assigned a unique identifier, known as a service tag. You can then use this tag to track usage of the service, for example for billing purposes.

Every conference event and participant event associated with the conference will include the service tag.

In most cases, if you are using service tags in your deployment you would obtain the information you require from the logs either from the Pexip Infinity Management API (for more information, see [Log output](#)), or from an [event sink](#). However, you can also search the [Administrator log](#) directly from the web-based Administrator interface (**History & Logs > Administrator Log**) using the service tag to track usage of the service.

Example

A service provider is using the Pexip Infinity Management API to create Virtual Meeting Rooms automatically on behalf of customers. They assign each new VMR a user-friendly **Service name** - this is the name that is presented to participants accessing the conference and appears in Skype for Business / Lync users' contact lists. They also assign each VMR a randomly generated UUID as the **Service tag**. The service provider then uses the Service tag UUID to track usage of each VMR in order to bill each customer appropriately for using their services.

Participant call tags

Every participant event associated with a conference can include a call tag that is specific to that participant.

The tag can be specified in client API requests and then used by app developers to correlate other API requests. The participant call tag is present in policy requests, participant updates to other participants, event sinks and the status/history databases for management API requests.

Note that the call tag is not displayed in the Administrator interface.

Automatically sending usage statistics


As part of our desire to continuously improve operation of the Pexip Infinity platform and Infinity Connect clients based on real-life customer usage data, we have enabled the ability for customers, at their own discretion, to automatically provide us with statistics relating to the usage and configuration of their deployment.

When [enabled at a platform level](#), relevant conference information (such as conference duration and number of participants, call duration and protocol) and selected configuration settings (such as version, license information, location and number of conferencing nodes) is sent automatically every 30 minutes to a third party service provider on behalf of Pexip.

When [enabled for individual Infinity Connect clients](#), information on events (such as use of conference controls, application errors, and settings such as bandwidth and device selection), is sent on occurrence of the event.

We will aggregate and analyze the information for each customer who has enabled these options, in order to better understand the ways our products are being used in production deployments. This in turn helps us determine the features and requirements of future software releases, which will ultimately provide a product that is of maximum benefit to you.

The analytics information we receive will not contain any identifying information (such as user names or passwords), and will not be shared with any other organizations. A full list of the data that will be sent to us at a platform level and by the Infinity Connect clients is given in [Information sent when usage reporting is enabled](#). For any more information, please contact your Pexip authorized support representative.

 When this option is enabled, either your Management Node must have a [web proxy configured](#), or your firewall must be configured to allow the Management Node to send outbound traffic over HTTPS.

Enabling and disabling automatic sending of platform usage statistics

The option to automatically send usage statistics is enabled or disabled during initial installation of the Management Node (when running the installation wizard).

To enable or disable the automatic submission of usage statistics after initial installation or upgrade:

1. On the Pexip Infinity Administrator interface, go to **Platform > Global Settings**.
2. From the **Reporting** section, select or deselect the **Automatically send deployment and usage statistics to Pexip** box as required.
3. Select **Save**.

Enabling Infinity Connect usage statistics

Individual Infinity Connect users can also control via their Infinity Connect application's **Settings** page whether to **Send anonymous statistics**.

Note that the **Automatically send deployment and usage statistics to Pexip** global setting on the Management Node must also be enabled in order to allow the Infinity Connect application to send usage statistics.

Information sent when usage reporting is enabled

Platform

When **Automatically send deployment and usage statistics to Pexip** is enabled, the following platform usage statistics are sent by the Management Node to Pexip's service provider:

Information sent
Common
The version of Pexip Infinity software running on the Management Node.
The Pexip Infinity software license ID.
Public IP address of the Management Node from which the report has been sent.
The name of the Management Node.
Call
Reason the call was disconnected.
Whether the call was to a Virtual Meeting Room, Virtual Auditorium, Virtual Reception, or Infinity Gateway.
Number of seconds this call leg was alive.
Vendor of the endpoint that placed the call.
Protocol used by the endpoint (SIP, H.323, WebRTC, etc).
Maximum bandwidth signaled.
Whether the participant was a Guest or Host.
IP address of the Conferencing Node that handled the media for the call.
IP address of Conferencing Node that handled the signaling for the call.
Whether the media stream was encrypted.
Unique identifier of this call (UUID).
Unique identifier of the conference instance (UUID).
The time at which the call was connected.
The time at which the call was disconnected.
Identifier of the system location of the Conferencing Node to which the call was connected.
Whether the call was incoming or outgoing.
The perceived overall quality of the call.
The type of license used by the call.
The number of licenses consumed by the call.
Audio/Video/Presentation (statistics for each type of stream, where present)
Whether the stream is audio, video or presentation.
The format used by the Conferencing Node to encode the media stream being sent to the endpoint.

Information sent

The quantity of data sent to the endpoint.

The display resolution of the image sent to the endpoint.

The total quantity of packets sent to the endpoint.

The total quantity of packets sent from the Conferencing Node but not received by the endpoint.

The format used by the Conferencing Node to decode the media stream being sent from the endpoint.

The quantity of data received from the endpoint.

The display resolution of the image received from the endpoint.

The total quantity of packets received by the Conferencing Node from the endpoint.

The total quantity of packets sent from the endpoint but not received by the Conferencing Node.

Conference

Cryptographic hash of the name of the Virtual Meeting Room, Virtual Auditorium, Virtual Reception or Infinity Gateway that was used.

The date and time that the first participant connected to the service.

The date and time that the last participant's call ended.

Number of seconds conference was active.

Whether the call was to a Virtual Meeting Room, Virtual Auditorium, Virtual Reception, or Infinity Gateway.

Total number of participants who connected to this conference instance.

The number of chat messages exchanged on this conference.

Platform (sent once a day for each system location)

Numeric identifier of the system location.

Number of nodes in that system location.

Configuration

Whether RTMP is enabled.

Whether SIP (TCP and TLS) is enabled.

Whether SIP TLS certificate verification mode is set to *On* or *Off*.

Whether WebRTC is enabled.

The URL to which OCSP requests will be sent.

The start value for the range of ports that all Conferencing Nodes use to send media.

The end value for the range of ports that all Conferencing Nodes use to send media.

The start value for the range of ports that all Conferencing Nodes use to send signaling.

The end value for the range of ports that all Conferencing Nodes use to send signaling.

Whether outbound calls are enabled.

Whether SSH is enabled.

Information sent

Whether OCSP will be used when checking the validity of TLS certificates.

The DSCP value for management traffic sent from the Management Node and Conferencing Nodes.

The number of minutes a browser session may remain idle before the user is logged out of the Pexip Infinity Administrator interface.

The URL to which incident reports will be sent (if enabled).

Whether HTTP access for external systems is enabled.

Whether H.323 is enabled.

Whether incident reporting is enabled.

Whether support for Pexip Infinity Connect and Mobile App is enabled.

The number of configured devices (including those that are not registered).

The number of configured services (Virtual Meeting Rooms + Virtual Auditoriums + Virtual Receptions).

Whether chat messages are enabled.

Whether SIP UDP is enabled.

Whether the next generation Infinity Connect web app is enabled.

Registration

The number of users registered over SIP, H.323 and WebRTC.

Infinity Connect clients

When **Send anonymous statistics** is enabled on an Infinity Connect client, the following information is sent by the client to Pexip's service provider:

Information sent**Registration**

When the client has failed to register, and the reason why.

When the client has registered, and the [Route calls via registrar](#) setting it will use.

When the user has intentionally unregistered the client.

Client configuration

Whether the client obtained its status information from the host server or Pexip Infinity, and whether it is using cached information.

Whether the camera was muted or unmuted when the client was opened.

When a user mutes or unmutes their camera from the home screen (prior to placing a call).

Whether the microphone was muted or unmuted when the client was opened..

When a user mutes or unmutes their microphone from the home screen (prior to placing a call).

When a user selects their speakers from the home screen.

When a user changes their display name.

Information sent

When an Android user denies permission to access their device's calendar.

Placing a call

When a user initiates a call by selecting an address from the **Recents** list.

When a user initiates a call by selecting a meeting from the **Calendar** list.

The bandwidth to be used for the call.

When the user has entered an invalid meeting PIN.

When the user has entered an invalid Virtual Reception extension.

Details of any errors that were not presented to the user.

Details of any errors that were presented to the user.

When a user rejoins a conference after a failed call by selecting the **Rejoin now** option.

When a call fails.

The version of Pexip Infinity that the client is connecting to.

When the client failed to connect to a host server.

Presentation

When the user attempts to share their screen but does not have the Pexip Screensharing Extension installed.

When the user starts and stops presenting content, and whether they have elected to share their screen or present files.

When the user starts and stops receiving content from another participant, and whether:

- the content is a screen share or files
- the content is received in HD

When the user elects to view a presentation in a separate window, and when they elect to close that window

Conference control

When the user has selected **Copy meeting link**

When the user locks or unlocks the meeting

When the user mutes or unmutes all Guests

When the user selects **Disconnects all participants**

When the user selects **Get media stats**

When the user adds a participant to the meeting.

When the user selects the option to **Send DTMF** to another participant

When the user has sent DTMF tones, and whether they were sent to another participant, or to the conference.

When the user sends or receives a chat message (note that the content of the message is not logged).

When the user selects **Show Info** for a particular participant.

When the user changes the role of a participant.

Information sent

When the user mutes or unmutes an individual participant.

When the user disconnects an individual participant.

When the user transfers a participant.

When the user's attempt to transfer a participant fails.

When the user mutes or unmutes their camera during a call.

When the user mutes or unmutes their microphone during a call.

When the user has been muted.

When the user escalates a presentation and control-only call to an audio call

When the user will be or has been disconnected.

Legacy Infinity Connect clients

When **Send anonymous statistics** is enabled on a legacy Infinity Connect client, the following information is sent by the client to Pexip's service provider:

Information sent

Indicates an attempt to place a call.

Indicates the call was connected.

Unique call identifier.

Properties of the type of client running Infinity Connect.

IP address of the client.

Width of the application window.

Height of the application window.

Screen width (maximum size available).

Screen height (maximum size available).

The bit depth of the color palette for displaying images.

The color resolution (in bits per pixel) of the screen.

Client/user language.

Whether the participant had audio-only media enabled.

Whether the participant had video (and audio) media enabled.

The requested connection bandwidth.

Indicates the participant initiated screen sharing / presenting.

Number of slides uploaded (when presenting).

Dial out participant role.

Dial out participant protocol.

Downloading a diagnostic snapshot

If you are experiencing issues with your Pexip Infinity service, you may be asked by your Pexip authorized support representative to provide them with a snapshot of your system to assist them in diagnosis of the issue.

To download the diagnostic snapshot:

1. On the Pexip Infinity Administrator interface, go to **Utilities > Diagnostic Snapshot**.
2. To download all available data, select **Download full snapshot**.
or
To download a subset of the snapshot, containing a specified time period:
 - a. Type in, or adjust the sliders to the start and end times (in terms of how many hours ago from the current time) of the diagnostic data to be downloaded. The number of hours available will vary depending on the logs available on the system.
 - b. If requested by your Pexip authorized support representative, select **Include diagnostic metrics from all Conferencing Nodes**.
If this is selected, then metrics from the Management Node and all Conferencing Nodes are included with the snapshot.
If this is not selected, then metrics from the Management Node only are included with the snapshot.
 - c. Select **Download limited duration snapshot**.Wait while the snapshot file is prepared — do not navigate away from the page until the file has been generated.
3. Follow your browser's prompts to save the file.

File contents

The diagnostic snapshot is a collection of logs, incident reports, and metrics, and therefore contains information such as IP addresses, conference names, aliases, Pexip Infinity configuration and system logs. Some of this information may be sensitive to your organization, so the snapshot should be saved and handled securely.

Note that Management Node metrics are always included in a snapshot. Conferencing Node metrics are included if you download a full snapshot or if you download a limited duration snapshot and have selected **Include diagnostic metrics from all Conferencing Nodes**.

File size

When each log reaches its maximum size, the oldest data is overwritten.

The maximum file size of each log in the snapshot is as follows:

- Administrator: 1 GB
- Support: 2 GB
- Internal developer-specific logs: 2 GB

The diagnostic snapshot also includes all available [incident reports](#):

- the full snapshot contains up to 28 days of incident reports, with a cap at 5 GB
- the limited duration snapshot contains incident reports from the selected number of hours.

Performing a network packet capture

The packet capture utility allows you to capture network traffic to and from Conferencing Nodes in one or more locations, and the Management Node. Packet captures can be helpful when diagnosing various network issues, for example problems with DNS, NTP and firewalls.

If you are experiencing issues with your Pexip Infinity service, you may be asked by your Pexip authorized support representative to provide them with one or more packet captures of your system to assist them in diagnosis of the issue.

To perform a packet capture:

1. From the Administrator interface, go to **Utilities > Packet Capture**.
2. Select the **System locations** where you want to run the packet capture and move them into the **Chosen System locations** list.
The packet capture will run against all of the Conferencing Nodes in the chosen locations. It will capture both interfaces if a Conferencing Node has dual network interfaces.
You can also choose the Management Node.

3. IPsec traffic is captured by default.
Clear the **Capture IPsec traffic** check box if you do not want to capture backplane IPsec traffic between Conferencing Nodes, in addition to regular user-facing traffic.
4. Encryption keys for SIP, H.323 and WebRTC traffic, and IPsec traffic if that is also being captured, are captured by default.
Clear the **Log encryption keys** check box if you do not want to capture encryption keys.
 - i The keys are captured in the log files so you'll typically need to download a [diagnostic snapshot](#) as well.
5. Select the **Duration** of the packet capture. It can be in the range 10-600 seconds.
6. Select **Start packet capture**.
 - i All previous packet captures are deleted when a new packet capture is started. Make sure that you have downloaded any previous packet capture files that you want to keep.

The capture will run for the specified duration. You cannot stop a packet capture after it has started.

A **Time remaining** countdown will display while the packet capture is running.
7. When the capture completes, the captured "pcap" files are listed at the bottom of the page.
The page may take a few seconds to sync. If all of the expected files aren't present after one minute then refresh the page.
8. You can **Download** or **Delete** any or all of the individual packet capture files.

Taking a manual packet capture from non-communicating nodes

If a Conferencing Node has lost communication with the Management Node, or if otherwise instructed by your Pexip authorized support representative, you may have to perform a manual capture via a direct SSH connection with that node:

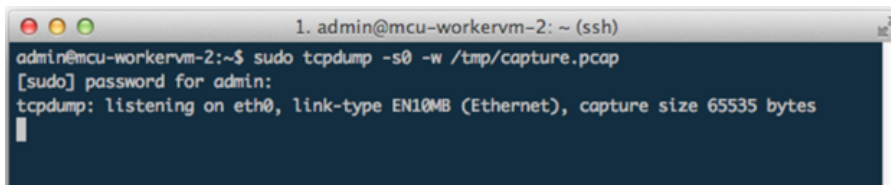
1. Connect to the Pexip node over SSH (using Putty, SecureCRT or another SSH client), logging in as user **admin**.
2. At the SSH command line, issue the following command, supplying the admin password when prompted.:

```
sudo tcpdump -s0 -w /tmp/capture.pcap
```

If the node has dual interfaces you need to add either the `-i nic0` or `-i nic1` switch to the tcpdump command, depending on which interface you want to capture.

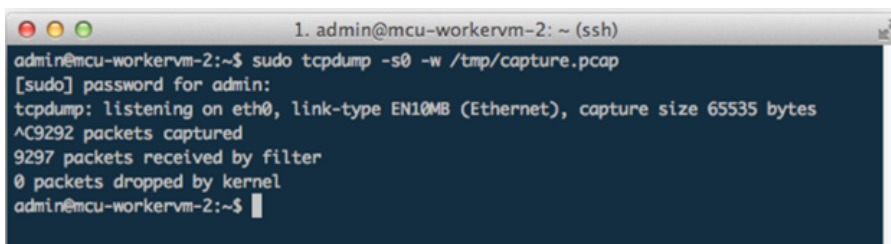
This starts a packet capture, storing the captured data in file `/tmp/capture.pcap` on the Pexip node.

3. While the capture is running, the following output can be observed:



```
1. admin@mcu-workervm-2: ~ (ssh)
admin@mcu-workervm-2:~$ sudo tcpdump -s0 -w /tmp/capture.pcap
[sudo] password for admin:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
```

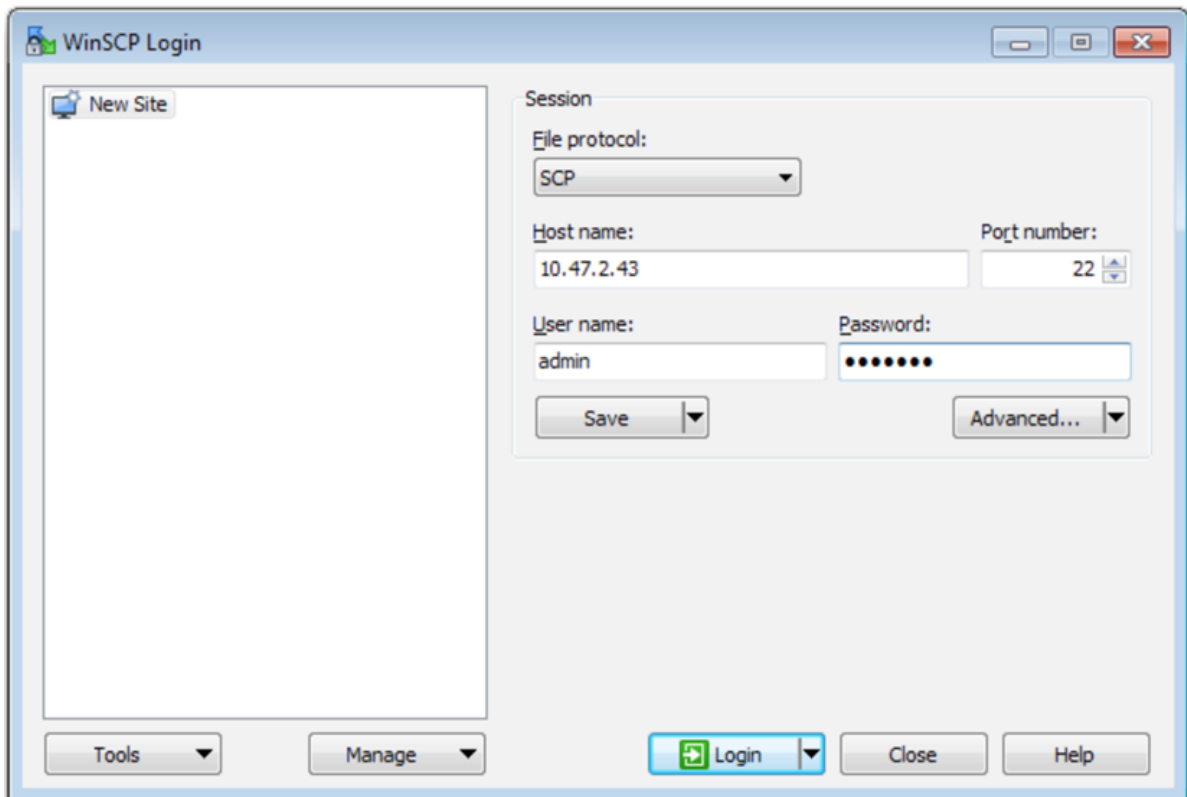
4. To stop the capture, press **Control + C** in the SSH window:



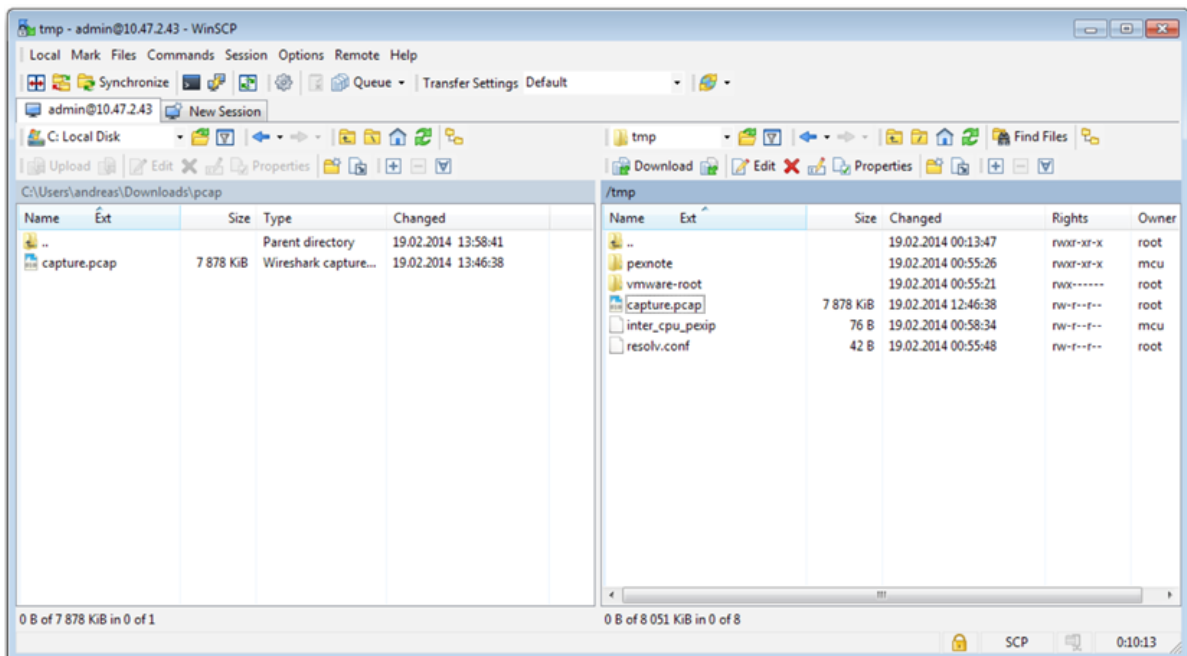
```
1. admin@mcu-workervm-2: ~ (ssh)
admin@mcu-workervm-2:~$ sudo tcpdump -s0 -w /tmp/capture.pcap
[sudo] password for admin:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
^C9292 packets captured
9297 packets received by filter
0 packets dropped by kernel
admin@mcu-workervm-2:~$
```

You can now close the SSH session.

5. To download the captured file `capture.pcap`:
 - a. Connect to the Pexip node with an SCP (Secure Copy) client, for example WinSCP — make sure to use SCP as the transfer protocol.
 - b. Log in as user **admin** and supply the admin password.



- c. Navigate to folder /tmp and download the capture file, named capture.pcap.



Viewing Conferencing Nodes

You can view details of all of your current Conferencing Nodes, including an interactive Conferencing Node status chart showing the load history for that node. You can also review a list of [historic events](#) (such as starting or stopping cloud bursting nodes) that have occurred on your Conferencing Nodes.

Viewing current Conferencing Nodes

To see a list of all Conferencing Nodes, go to **Status > Conferencing NodeS**.

Each Conferencing Node has the following information available:

Field	Description
Name	The name of the Conferencing Node. Click on the name to view detailed status information.
Role	Indicates the node's role — either a Transcoding Conferencing Node or a Proxying Edge Node.
IPv4 Address	The primary IP address of the Conferencing Node.
Secondary address *	The optional secondary interface IPv4 address of the Conferencing Node.
Static NAT address *	The optional static NAT address of the Conferencing Node.
System location	The physical location of the Conferencing Node.
Deployment status	<p>Deployment succeeded indicates that the Conferencing Node has been successfully deployed and is in contact with the Management Node.</p> <p>Not deployed indicates that there has not yet been any contact from the Conferencing Node.</p>
Maintenance mode	Indicates whether the node is currently in maintenance mode .
Version	The version number of the Pexip Infinity software that is currently installed on this Conferencing Node.
Last contacted	<p>The date and time that the Conferencing Node last contacted the Management Node. In typical deployments, configuration replication is performed approximately once per minute. However, in very large deployments (more than 60 Conferencing Nodes), configuration replication intervals are extended, and it may take longer for configuration changes to be applied to all Conferencing Nodes (the administrator log shows when each node has been updated).</p> <ul style="list-style-type: none"> • Green text indicates that there are no issues. • Orange text indicates that no contact has been made within the last 2 expected replication intervals (typically within the last 3–5 minutes). • Red text indicates that no contact has been made for some time — typically none within the last 5 minutes, but note that this replication interval is extended in very large deployments.
Last updated	The last time that the Conferencing Node's configuration was updated.
Maximum connections	For a Transcoding Conferencing Node, the initial table shows an estimate of the total number of simultaneous high-definition (HD) 720p video calls this node can handle.†
Audio connections *	
SD connections *	After you have selected an individual transcoding node to view, you can also review the estimated SD, HD, Full HD and audio call capacity.
HD connections *	
Full HD connections *	
Proxy audio connections *	For a Proxying Edge Node, the initial table shows an estimate of the total number of proxied video calls this node can handle. After you have selected an individual proxying node to view, you can also review the estimated proxying capacity for audio calls.
Proxy video connections *	
Media load	<p>An estimate of how much of its total capacity the Conferencing Node is currently using.†</p> <p>Note that when a Conferencing Node is in maintenance mode, it reports a media load of 100%. This is to indicate that there is no current capacity available.</p>

Field	Description
Call count *	The number of signaling connections that the node is currently handling.
System memory	Information about the Virtual Machine hosting this Conferencing Node, including: <ul style="list-style-type: none"> The amount of RAM configured on this Virtual Machine. The configured number of virtual CPUs. The underlying CPU model. The highest CPU instruction set supported by this Virtual Machine. The hypervisor on which this Virtual Machine is running.
Number of vCPUs	
CPU model	
CPU instruction set	
Hypervisor	
Usage graph *	Shows media usage % statistics for this Conferencing Node for the last 4 weeks. <div data-bbox="444 520 940 806"> <p>Status history (interactive)</p> </div> <p>You can click on the graph and then scroll to zoom in and out, and you can drag the timebox to the left or right to move forwards or backwards in time.</p>

* Only displayed when you have selected an individual Conferencing Node to view.

† Capacity estimates assume that only calls of that media type are being handled. For more information on server capacity, see [Capacity planning](#).

Viewing historic Conferencing Node events

Go to **History & Logs > Conferencing Node History** to see all of the events that have occurred on your Conferencing Nodes. These events include:

- Adding or removing a node.
- Placing a node into or out of maintenance mode.
- Starting up or stopping a cloud bursting node.

Viewing system location status

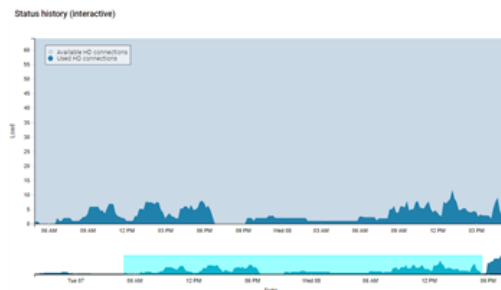
To see a list of all system locations currently configured, and their media usage statistics, go to **Status > Locations**.

i The [Live view](#) page (**Status > Live View**) lets you review current and historic usage charts showing a breakdown of participants by location, protocol, license type and the different conference types being hosted.

The following information about each location is available:

Field	Description
Name	The name of the location.
Maximum connections	The initial table shows an estimate of the maximum number of connections (in terms of HD calls for a location containing transcoding nodes, and video calls for a location containing proxying nodes) that can be supported by all of the Conferencing Nodes assigned to this location.
Audio connections *	After you have selected an individual location to view, you can also review the estimated SD, HD, Full HD and audio call capacity for a location containing transcoding nodes, or the estimated proxying capacity for audio and video calls for a location containing proxying nodes.
SD connections *	
HD connections *	
Full HD connections *	
Proxy audio connections *	
Proxy video connections *	
Media load	An estimate of how much of the location's total capacity is currently being used.
Transcoding resources	The system location to handle media transcoding for calls (signaling) received in, or sent from, this location.
Usage graph *	Shows media usage statistics for this location for the last 7 days. It indicates the resources that are actually being consumed (dark blue area) over time, relative to the maximum possible media load (light blue area). The load is shown in terms of HD calls or proxied video calls as appropriate for the nodes in that location (see Call types and resource requirements for more information).

The maximum possible media load typically stays consistent over time, but can vary if Conferencing Nodes are added to, or removed from, the location.



You can click on the graph and then scroll to zoom in and out, and you can drag the timebox to the left or right to move forwards or backwards in time.

* Only displayed when you have selected an individual location to view.

If a location has no associated Conferencing Nodes (for example, after the location has just been created), the location reports a media load of 100%. This indicates that there is no current capacity available in that location.

Viewing cloud bursting status

You can view the current status of your overflow nodes and locations, and view a history of all events that have been applied to overflow nodes.

Viewing current status


Go to **Status > Cloud Bursting** to see an overview of the media load of your principal locations (that contain your "always-on" Conferencing Nodes), and whether your overflow nodes and locations are in use.

- Any issues relating to your cloud bursting deployment will also be shown on this page.
- The list of principal locations only includes those locations that are configured with a **Primary overflow location** that contains bursting nodes.
- An **approaching threshold** message is displayed in the **Available HD** connections column for the principal locations when the number of available HD connections is less than or equal to the bursting threshold plus two.
This message changes to **bursting threshold reached** when the number of available HD connections is less than or equal to the bursting threshold (and therefore overflow nodes are started up).
- You can manually start any overflow nodes by selecting **Start** for the required node (the **Start** option is in the final column of the **Cloud overflow nodes** table).
- The status page dynamically updates every 15 seconds.

Viewing historic events

Go to **Status > Conferencing Node History** to see all of the events (stop, start or running) that have been applied to overflow Conferencing Nodes and, where appropriate, the reason why the event was applied (for example if a node was shut down as there was no longer a need for the extra capacity).

Viewing alarms

When there are active alarms on your Pexip Infinity deployment, a flashing blue triangle  appears at the top right of each page of the Administrator interface. To view details of the current alarms, click on this icon or go to the **Alarms** page (**Status > Alarms**).

- Alarms remain in place for as long as the issue exists. After the issue has been resolved (for example, if a conference ends, therefore freeing up licenses) the associated alarm will automatically disappear from the **Alarms** page.
- Multiple instances of the same type of alarm can be raised. For example if two Conferencing Nodes are not correctly synchronized to an NTP server, you will see an alarm for each node.
- You can select individual alarms and view the associated documentation (this guide) for suggested causes and resolutions.

The **History & Logs > Alarm History** page shows the details of all historic alarms including the severity level, and the time the alarm was raised and lowered.

An alarm is raised in each of the following situations:

Alarm	ID	Logged as Alarm =	Level	Cause	Suggested resolutions
The Management Node does not have a TLS certificate	20	tls_certificate_missing_management	Critical	The Management Node has no associated TLS certificate.	Upload a TLS certificate and associate it with the Management Node.
A Conferencing Node does not have a TLS certificate	9	tls_certificate_missing	Critical	A Conferencing Node has no associated TLS certificate.	<p>Upload a TLS certificate and associate it with the Conferencing Node.</p> <p>Alternatively, and if appropriate for your deployment, associate an existing certificate with your Conferencing Node. When doing this, the existing certificate should already contain a SAN (Subject Alternative Name) that matches your Conferencing Node's FQDN.</p> <p>See Managing a node's TLS server certificate for more information.</p>
CPU instruction set not supported	10	cpu_not_supported	Critical	<p>A Conferencing Node has gone into maintenance mode because it was deployed on a server with an unsupported processor instruction set (e.g. SSE4.1).</p> <p>This could also be caused by setting the EVC mode on a VMware cluster to too low a level, such as Westmere.</p>	Deploy the Conferencing Node on a server with AVX or later.
Eventsink Reached Maximum Backoff	36	eventsink_maximum_backoff	Critical	An event cannot be delivered to an event sink, and the system has reached its retry timeout limit.	See Troubleshooting event sink failures .
Eventsink Reached Maximum Concurrent POSTs	37	eventsink_maximum_posts	Critical	More than the configured Maximum number of background POSTs events (default 1000) are queued for an event sink but have not been sent.	See Troubleshooting event sink failures .

Alarm	ID	Logged as Alarm =	Level	Cause	Suggested resolutions
NTP not synchronized	11	ntp_not_synchronised	Error	A node has failed to synchronize with the configured NTP servers.	Ensure that NTP is enabled on the Management Node, and that NTP servers are assigned to, and accessible from, each location. See Syncing with NTP servers for more information.
Configuration not synchronized	18	configuration_sync_failure	Error	This alarm is raised if the Conferencing Node status "Last contacted" time has not been updated within the last 2 expected replication intervals (typically no contact within the last 3 minutes).	<p>In typical deployments, configuration replication is performed approximately once per minute. However, in very large deployments (more than 60 Conferencing Nodes), configuration replication intervals are extended, and it may take longer for configuration changes to be applied to all Conferencing Nodes (the administrator log shows when each node has been updated).</p> <p>If configuration synchronization fails this may indicate network connectivity or routing issues between the Management Node and the Conferencing Node, which could be due to a malfunction or misconfiguration of devices such as routers or firewalls etc.</p> <p>Ensure that all of the appropriate Pexip nodes are fully routable to each other in both directions. See General network requirements.</p>
MS Exchange Connection Failure	22	scheduling_connection_failure	Error	The Management Node cannot connect to the Exchange server.	Check that the details entered in the EWS URL (System > VMR Scheduling For Exchange IntegrationS) are correct and the Exchange server is online.
Automatic backup upload failed	25	autobackup_upload_failed	Error	The Management Node cannot connect to the FTP server to upload a backup file.	Check that the Upload URL (supported schemes are FTPS and FTP) and the Username and Password credentials of the FTP server are correct (Utilities > Automatic Backups) and that the Management Node can reach the FTP server.
LDAP sync failed	28	ldap_sync_failure	Error	An LDAP template synchronization process has failed. This alarm duplicates the information shown for the error listed at Status > LDAP Sync.	<p>See Troubleshooting LDAP server connections for help with resolving LDAP connection issues.</p> <p>The alarm is lowered when you resync the template (although it will get re-raised if the issue has not been resolved).</p>

Alarm	ID	Logged as Alarm =	Level	Cause	Suggested resolutions
Pexip Private Cloud gateway failure	34	pss_gateway_failure	Error	There was a problem connecting to the Pexip Private Cloud via the configured gateway (Platform > Global Settings > Pexip Private Cloud > Gateway URL).	<ul style="list-style-type: none"> Check that the Gateway URL is in https:// format. Check that the Customer ID and Authentication token are correct.
License limit reached	2	licenses_exhausted	Warning	<p>A Conferencing Node is unable to accept a call because there are not enough concurrent licenses available on the system at this time. For more information, see Pexip Infinity license installation and usage.</p>	<ul style="list-style-type: none"> Wait until one or more of the existing conferences have finished and the licenses have been returned to the pool. Contact your Pexip authorized support representative to purchase more licenses. <p>Note that when a license subsequently becomes available (e.g. because a participant leaves a conference, or because the administrator adds more licenses), the alarm is not cleared immediately; the alarm is cleared after the next participant successfully joins a conference.</p>
Licenses expiring	3	licenses_expiring	Warning	One or more of your licenses is due to expire within the next 60 days.	Contact your Pexip authorized support representative to renew your licenses.
Call capacity limit reached	1	capacity_exhausted	Warning	<p>A call has not been accepted because all Conferencing Nodes that are able to take the media for this call are at capacity. It could be either Proxying Edge Nodes or Transcoding Conferencing Nodes that are out of capacity.</p> <p>Note: to understand how often this issue is occurring in your deployment, search the Administrator log for "out of proxying resource" or "out of transcoding resource".</p> <p>This alarm clears either when an existing call is disconnected or the next time a new call is successfully placed.</p>	<ul style="list-style-type: none"> Deploy more Conferencing Nodes in either the proxying or transcoding location as appropriate. Move existing Conferencing Nodes onto more powerful servers. Allocate more virtual CPUs for Conferencing Nodes on existing servers (if there are sufficient CPU cores). Note that the Conferencing Node will have to be rebooted for this to take effect. Configure each location with a primary and secondary overflow location. If a call is received in a location that contains Proxying Edge Nodes, that location must be configured with a Transcoding location that contains your Transcoding Conferencing Nodes. <p>Note that some types of call consume more resources than other calls. Thus, for example, if you are at full capacity and an audio-only call disconnects, there may still not be sufficient free resource to connect a new HD video call.</p>

Alarm	ID	Logged as Alarm =	Level	Cause	Suggested resolutions
Management Node limit reached	5	management_node_exhausted	Warning	The Management Node does not have sufficient resources for the current deployment size (number of Conferencing Nodes).	<p>Increase the amount of RAM and the number of virtual CPUs assigned to the Management Node.</p> <p>See the recommended hardware requirements in Server design recommendations.</p>
Trusted CA certificates expiring	6	trustedca_expiring	Warning	One or more of your trusted CA certificates is due to expire within the next 30 days, or has already expired.	Obtain and upload an updated certificate for the certificate authority.
TLS certificates expiring	7	tls_certificate_expiring	Warning	One or more of your TLS certificates is due to expire within the next 30 days, or has already expired.	Obtain and upload an updated TLS certificate. You may also need to delete the old certificate.
Incomplete TLS certificate chains	8	tls_certificate_chains	Warning	A TLS certificate has an incomplete chain of trust to the root CA certificate.	Obtain and upload the appropriate chain of intermediate CA certificates to the Management Node (the certificate provider normally provides the relevant bundle of intermediate CA certificates).
Syslog server inaccessible	4	syslog_inaccessible	Warning	A syslog server has been configured to use TCP or TLS but either is not responding to contact requests, or the connection has dropped.	<ul style="list-style-type: none"> • Check your network connectivity. • Check that the syslog server is running.

Alarm	ID	Logged as Alarm =	Level	Cause	Suggested resolutions
Connectivity lost between nodes	19	connectivity_lost	Warning	<p>Communication to a Pexip Infinity node has been lost.</p> <p>More information</p> <p>When a connection is lost, Pexip Infinity tries to contact the node every 5 seconds until the connection is re-established. In large deployments with many connectivity failures, it attempts to re-establish connections to a maximum of 10 nodes at a time.</p> <p>Intermittent short-lived "Connectivity lost between nodes" alarms may be an indication of an unreliable network.</p> <p>These alarms may be raised for a short period of time — as expected behavior — if a node is placed into maintenance mode and Proxying Edge Nodes need to establish new connectivity paths.</p> <p>They may occur during initial deployment or an upgrade, and is also expected behavior. They automatically clear as each node is upgraded to the new software version, has restarted and is ready to handle calls.</p> <p>When restricted routing for Proxying Edge Nodes is enabled, you may see these alarms (and is expected behavior):</p> <ul style="list-style-type: none"> When deploying proxying nodes if, for example, the location containing your proxying nodes is configured with a Transcoding location that doesn't yet contain any transcoding nodes. In this case the alarm will be lowered when transcoding nodes are deployed in that location. If all of the nodes in the edge location's Transcoding location (and any configured media overflow locations) are in maintenance mode. This applies even if all of the proxying nodes are also in maintenance mode. 	Check network connectivity and routing as for "Configuration not synchronized" above, or in the case of a software upgrade, wait for the upgrade process to complete.

Alarm	ID	Logged as Alarm =	Level	Cause	Suggested resolutions
Hardware instability detected	21	irregular_pulse	Warning	Pexip Infinity has detected that the underlying VM infrastructure has paused the Pexip virtual machine. This is usually indicative of over-committed hardware, which we do not support. Pexip Infinity is a real time system and requires dedicated access to the underlying CPU and RAM resources of the hardware host.	<p>Ensure that the Management Node and all Conferencing Nodes have dedicated access to their own RAM and CPU cores.</p> <p>See the recommended hardware requirements in Server design recommendations.</p>
CPU instruction set is deprecated	23	cpu_deprecated	Warning	<p>The node is deployed on a server that is not using the AVX or later CPU instruction set (e.g. if it uses SSE4.2).</p> <p>This alarm is raised when a Conferencing Node restarts and is automatically cleared after 48 hours.</p>	Deploy the Conferencing Node on a server with AVX or later.
Hardware IO (input/output) instability detected	24	io_high_latency	Warning	Pexip Infinity has recently detected consistent read latency greater than 100ms or write latency greater than 400ms.	<ul style="list-style-type: none"> Avoid having multiple VMs using the same physical hard drive. Check the hard drive for failures.
VOIP scanner resistance has detected excessive incorrect aliases being dialed in a short period	26	possible_voip_scanner_ips_blocked	Warning	Pexip Infinity's VOIP scanner resistance has detected excessive incorrect aliases being dialed in a short period, and has temporarily blocked access attempts from the suspected VOIP scanner IP addresses.	See the administrator log for details of the calls.
PIN brute force resistance has detected excessive incorrect PIN entry attempts in a short period	27	service_access_quarantined	Warning	Pexip Infinity's PIN brute force resistance has detected excessive incorrect PIN entry attempts in a short period, and has temporarily blocked access attempts to one or more conferencing services.	See the administrator log for details of the calls.
Pexip Private Cloud configuration failed	33	pss_config_failure	Warning	A configuration change to a Pexip Smart Scale location could not be applied.	Check that there are no issues communicating with the PPC gateway. Otherwise, contact your Pexip authorized support representative.
Scheduled maintenance event (freeze)	39	scheduled_maintenance_event_freeze	Warning	A scheduled maintenance event in Microsoft Azure has been detected.	No action is required.
Scheduled maintenance event (redploy)	40	scheduled_maintenance_event_redeploy	Warning		Any Conferencing Node running on the affected VM is automatically placed into maintenance mode until the event completes.
Scheduled maintenance event (preemption)	41	scheduled_maintenance_event_preempt	Warning		

Cloud bursting alarms

The following alarms may be raised in relation to issues with dynamic cloud bursting. See [Dynamic bursting to a cloud service](#) for more information about resolving these alarms.

Alarm	ID	Logged as Alarm =	Level	Cause	Suggested resolutions
Not authorized to perform this operation	15	bursting_unauthorized_instance_failure	Error	Pexip Infinity is not authorized to view instance data or to start and stop instances in the cloud service.	For AWS, ensure that an appropriate policy document is configured in AWS and is attached to the user that is being used by the Pexip platform.
	16	bursting_unauthorized_region_failure			For Azure, check your Active Directory (AD) application and its associated role/permissions. For GCP, check your service account and its associated role/permissions.
Authentication failure while trying to communicate with the cloud provider	17	bursting_authentication_failure	Error	Pexip Infinity cannot sign in to the cloud service.	Check your cloud bursting settings in Platform > Global Settings > Cloud Bursting: <ul style="list-style-type: none"> For AWS, check that the Access Key ID and Secret Access Key match the User Security Credentials for the user you added within Identity And Access Management in the AWS dashboard. For Azure, check that your subscription, client and tenant IDs and secret key are correct for your Active Directory application. For GCP, check that your configured GCP project ID, service account ID and private key are correct for your GCP service account.
Cloud bursting process encountered an unexpected error	12	bursting_error	Error	Pexip Infinity encountered an unexpected error while managing the cloud overflow nodes.	Check the status of your cloud bursting nodes within Pexip Infinity (Status > Cloud Bursting) and of your instances within your cloud provider. Also check administrator and support log messages that are tagged with a log module name of administrator.alarm to see additional error message information.

Alarm	ID	Logged as Alarm =	Level	Cause	Suggested resolutions
Cloud-bursting node found, but no corresponding Conferencing Node has been configured	13	bursting_missing_pexip_node	Warning	This occurs when Pexip Infinity detects a bursting instance with a tag matching your system's hostname but there is no corresponding Conferencing Node configured within Pexip Infinity.	This message can occur temporarily in a normal scenario when deploying a new Conferencing Node and you have set up the VM instance in your cloud provider but you have not yet deployed the Conferencing Node in Pexip Infinity. In this case, the issue will disappear as soon as the Conferencing Node is deployed.
A location contains cloud bursting nodes, but no other locations are using it for overflow	14	bursting_no_location_overflow	Warning	A location contains some cloud overflow nodes, but no other locations are using it as an overflow location.	Set the location containing the cloud overflow nodes as the Primary overflow location of the locations containing your "always on" Conferencing Nodes.

One-Touch Join alarms

The following alarms may be raised in relation to issues with One-Touch Join:

Alarm	ID	Logged as Alarm =	Level	Instance	Cause	Suggested resolutions
OTJ Google Gatherer Error	29	mjax_google_gatherer_failure	Error	Google Connection Test Failure	The connection test to Google Workspace has failed. This could be because your service account credentials are incorrect.	Check your service account details, specifically the service account email and private key.
				Google Room Connection Failure	OTJ has been unable to connect to one of the rooms you have specified. This could be because the room is misconfigured within Google Workspace.	Check the steps to set up a new room. Is the room resource email correct? Has it been shared with the service account?
OTJ Exchange Gatherer Error	30	mjax_exchange_gatherer_failure	Error	Exchange Connection Test Error	The connection test to Exchange has failed. This could be because your service account credentials are incorrect.	Check your Exchange service account username and password.
				Exchange OAuth Error	OTJ is unable to use OAuth to sign into Exchange.	Check your OAuth credentials.
				Exchange Room Connection Error	OTJ is unable to connect to the room specified in the alarm description. This could be because the room is misconfigured.	Check the room has been correctly set up.

Alarm	ID	Logged as Alarm =	Level	Instance	Cause	Suggested resolutions
OTJ Endpoint Configurator Error	31	mjsx_endpoint_configurator_failure	Error	Endpoint Misconfigured Error	The OTJ endpoint does not have a username and password configured, and there is no default username and password.	Provide a username and password for the OTJ endpoint or for the associated OTJ profile.
				Endpoint Request Error	OTJ is unable to connect to the endpoint.	Check that the endpoint is configured correctly.
				Endpoint Non-200 Status Code	The endpoint returns a non-200 status code.	Check the status code that is given in the logs. This is likely a configuration error with the endpoint.
OTJ Meeting Processor Failure	32	mjsx_meeting_processor_failure	Error	Meeting Processor Rendering Error, Template Error or Runtime Error	The meeting processing rule could not extract a meeting alias.	Check and edit the rule using the test tool.
OTJ Poly Endpoint Error	35	mjsx_poly_failure	Error	Poly Endpoint Not Polled	A Poly endpoint that has Raise alarms enabled has not made contact with the OTJ calendaring service within the last 10 minutes.	<p>Ensure that the configuration for endpoint on Pexip Infinity and on the endpoint itself is correct, in particular that the username and password configured on both match.</p> <p>Ensure that the endpoint is showing as registered to the calendaring service. Restart the endpoint.</p>

Alarm	ID	Logged as Alarm =	Level	Instance	Cause	Suggested resolutions
OTJ Webex Failure	38	mjsx_webex_failure	Error	Webex Endpoint failed with <Webex error message>	<p>One-Touch Join has received an error from Webex when attempting to send the request. Examples of these messages include:</p> <ul style="list-style-type: none"> Cloud Calendar is configured Device has not registered as an XAPI provider Webex Request Error No response received from request The server, while acting as a gateway or proxy, received an invalid response from the upstream server it accessed in attempting to fulfill the request 	<p>The resolution will depend on the issue, for example:</p> <ul style="list-style-type: none"> Disable the cloud calendar. Check that the Device ID is correct. Confirm that the correct ports are open, and that Pexip Infinity can reach Webex. Confirm that the endpoint is switched on and connected to the internet, and that Webex can reach it.
				Webex OAuth Error	We are unable to get an access token for your webex integration.	Sign in with the webex service account again.
				Webex Configuration Error	A webex endpoint is configured but there is no webex integration configured on the OTJ profile.	Make sure you configure a webex integration on the OTJ profile.

Viewing login history

To see a record of attempts to log in to the Pexip Infinity Administrator interface, go to **History & Logs > Login History**.

The system displays the 100 most recent login attempts. It displays successful and failed attempts, and for each attempt shows the time of the attempt, the username that was submitted, and the address and port of the remote client.

Note that you are taken directly to the **Login History** page after logging in to the Pexip Infinity Administrator interface if the system has been configured with a login banner and certificate-based authentication has been enabled.

About the support log

The Management Node collates the logs from itself and all Conferencing Nodes and compiles them into the support log.

The support log records all events occurring across the Pexip Infinity deployment, including:

- configuration changes on the system
- conferences starting and ending
- participants joining and leaving conferences
- participants presenting in conferences
- SIP, H.323 and BFCP signaling
- device registration requests
- DNS lookups.

Sensitive information, such as the content of chat messages, is not logged.

Log timestamps always use UTC.

The information in the support log is retained during reboot and upgrade.

 You can also configure Pexip Infinity to use an external [syslog server](#). This allows you to use the syslog server's associated tools to collate and process the log entries to obtain specific information about real-time events, tailored to your requirements.

For more information on the content of the support log, see [Log output](#).


Viewing the support log

To view the support log, go to **History & Logs > Support Log**. If required, you can export the support log to a text file by selecting **Download**.

The log appears in the format:

```
syslog_time system originating_time level name details
```

where:

Field	Description
<code>syslog_time</code>	In the format: <code>year-month-dayThour:minute:second.millisecondUTC_offset</code> The time that the event was logged by syslog on the originating system.
<code>system</code>	The IP address or host name of the system that sent the log message.
<code>originating_time</code>	In the format: <code>year-month-day hour:minute:second,millisecond</code> The time at which the event occurred on the originating system.
<code>level</code>	The severity of the event. The levels, in order of increasing severity, are: <ul style="list-style-type: none">• DEBUG• INFO• WARNING• ERROR  In the Pexip Infinity Administrator interface, warnings and errors are highlighted with an orange or blue background respectively.
<code>name</code>	The system module producing the log output.
<code>details</code>	Information about the event, as a series of name=value pairs.

Searching the support log

You can filter the entire support log to only show messages that contain a particular string (for example, an IP address or alias) using the search box at the top left of the web page.

The search box also supports BRE syntax [regular expressions](#) (regex), so for example, you could search for:

- `^2017-10-07T09:[34][[:digit:]]`: to limit the displayed log messages to those emitted between 09:30:00 and 09:49:59 UTC on 7 October 2017
- `.*"Registration added".*"sip:alice@example.com"` to find all instances of "Registration added" messages for the SIP alias `alice@example.com`
- `.*\("error"\|administrator.alarm\).*` to review error and warning alarms that have been raised or lowered.

Note that the results are not paginated.

Summarizing support log messages

You can select **Log summary** to generate a condensed view of the messages in the support log, showing a summary of the call signaling — similar to traditional ladder diagrams, but text based — for the current search/filtered log messages, such as a filter on a specific call ID.

File size

The support log has a maximum size of 2 GB, after which it will be overwritten, starting with the oldest entries. Up to 10 MB is available in a single page view on the Pexip Infinity Administrator interface; you can navigate through additional log pages using the buttons at the bottom left of the page. (Note that each page is a separate file, so the first page, which is the end of the file, may not be full.)

About the administrator log

The administrator log is a subset of the [support log](#). It contains information about events occurring during the normal use of the system which may be of interest to a system administrator, notably:

- Conferencing Node deployment and communication status
- configuration changes on the system
- conferences starting and ending
- participants joining and leaving conferences
- participants presenting in conferences
- device registrations
- VMR imports
- system backups
- break-in prevention policy triggers.

Sensitive information, such as the content of chat messages, is not logged.

Log timestamps always use UTC.

The information in the administrator log is retained during reboot and upgrade.


Viewing the administrator log

To view the administrator log, go to **History & Logs > Administrator Log**. If required, you can export the administrator log to a text file by selecting **Download**.

The log appears in the format:

```
syslog_time system originating_time level name details
```

where:

Field	Description
<code>syslog_time</code>	In the format: <code>year-month-dayThour:minute:second.millisecondUTC_offset</code> The time that the event was logged by syslog on the originating system.
<code>system</code>	The IP address or host name of the system that sent the log message.
<code>originating_time</code>	In the format: <code>year-month-day hour:minute:second,millisecond</code> The time at which the event occurred on the originating system.
<code>level</code>	The severity of the event. The levels, in order of increasing severity, are: <ul style="list-style-type: none">• DEBUG• INFO• WARNING• ERROR  In the Pexip Infinity Administrator interface, warnings and errors are highlighted with an orange or blue background respectively.
<code>name</code>	The system module producing the log output.
<code>details</code>	Information about the event, as a series of name=value pairs.

Searching the administrator log

You can filter the entire administrator log to only show messages that contain a particular string (for example, an IP address, alias or [service tag](#)) using the search box at the top left of the web page.

The search box also supports BRE syntax [regular expressions](#) (regex), so for example, you could search for:

- `^2017-10-07T09:[34][[:digit:]]`: to limit the displayed log messages to those emitted between 09:30:00 and 09:49:59 UTC on 7 October 2017
- `.*"Registration added".*"sip:alice@example.com"` to find all instances of "Registration added" messages for the SIP alias `alice@example.com`
- `.*\("error"\|administrator.alarm\).*` to review error and warning alarms that have been raised or lowered.

Note that the results are not paginated.

For more information on the content of the administrator log, see [Log output](#).


File size

The administrator log has a maximum size of 1 GB, after which it will be overwritten, starting with the oldest entries. Up to 10 MB is available in a single page view on the Pexip Infinity Administrator interface; you can navigate through additional log pages using the buttons at the bottom left of the page. (Note that each page is a separate file, so the first page, which is the end of the file, may not be full.)

Log output

The Pexip Infinity Management Node collates the logs from itself and all Conferencing Nodes and compiles these into the support log and the administrator log (which is a subset of the events in the support log).

To view these logs from the Pexip Infinity Administrator interface, go to **History & Logs > Support Log** or **History & Logs > Administrator Log**. If required, you can export the logs to a text file by selecting **Download**.


 You can also use a syslog server to collate logs remotely.

Log timestamps always use UTC.

The log appears in the format:

```
syslog_time system originating_time level name details
```

where:

Field	Description
<code>syslog_time</code>	In the format: <code>year-month-dayThour:minute:second.millisecondUTC_offset</code> The time that the event was logged by syslog on the originating system.
<code>system</code>	The IP address or host name of the system that sent the log message.
<code>originating_time</code>	In the format: <code>year-month-day hour:minute:second,millisecond</code> The time at which the event occurred on the originating system.
<code>level</code>	The severity of the event. The levels, in order of increasing severity, are: <ul style="list-style-type: none"> • DEBUG • INFO • WARNING • ERROR  In the Pexip Infinity Administrator interface, warnings and errors are highlighted with an orange or blue background respectively.
<code>name</code>	The system module producing the log output.
<code>details</code>	Information about the event, as a series of name=value pairs.

Administrator log system modules

The administrator log contains information about events occurring during the normal use of the system which may be of interest to a system administrator.

The following list shows typical events that are shown in the administrator log. In addition to these, logs may also contain one-off events, such as bulk configuration import and incident reports.

For the sake of brevity, the examples shown below feature only the `name` and `details` sections of the log message.

administrator.system

Logs when the Pexip Infinity application has started on this Conferencing Node. Also logs web-based activities such as:

- import of service configuration
- deployment of a Conferencing Node
- configuration synchronization towards Conferencing Nodes
- liveness status of Conferencing Nodes and connectivity paths between nodes (either a direct path, or via other nodes in some cases for Proxying Edge Nodes)

Examples

```
Name="administrator.system" Message="Starting Infinity application." Pid="6361" Version="15 (34498.0.0) "
```

```
Name="administrator.system" Message="Deploying conferencing node virtual machine." Address="192.168.0.2"
```

```
Name="administrator.system.configuration" Message="Conferencing node configuration updated." Node="conferencingnode1"
```

```
Name="administrator.system.connectivity" Message="Unable to contact node." Src-Node="192.168.0.5" Node="192.168.0.2"
```

```
Name="administrator.system.connectivity" Message="Connectivity re-established." Src-Node="192.168.0.5" Node="192.168.0.2" Path="Direct" Last-Reported="Wed Mar 22 23:13:50 2017"
```

```
Name="administrator.system.connectivity" Message="Connectivity established." Src-Node="10.47.2.17" Node="10.47.3.46" Path="10.47.3.4, 10.47.3.7" (in this last example there is an indirect connectivity path between the source node and the target node, which is 10.47.2.17 > 10.47.3.4 > 10.47.3.7 > 10.47.3.46)
```

administrator.configuration

This module logs when:

- configuration is added or changed
- a user logs in or logs out of the system, or fails to log in
- a participant is manually dialed into a conference
- license requests are processed

Messages

```
Configuration added  
Configuration changed  
Configuration deleted  
User logged in  
User logged out  
User login failed  
Initiated dial command  
License activation request stored for offline processing  
Processed license request
```

Parameters

Parameter	Definition
User	The user who made the change.
Type	The resource that was changed/added.
ID	The ID of the resource that was changed/added.
Fields	The fields in the resource that were changed. Applies to "Configuration changed" logs only.
Remote-Address	The IP address of the system from which the request was received.
Remote-Port	The port on the system from which the request was received.

Examples

```
Name="administrator.configuration" Message="User login failed" User="admins" Remote-Address="192.0.2.0" Remote-Port="53991"
```

```
Name="administrator.configuration" Message="Configuration added" User="scheduling-service" Type="Scheduled Alias" ID="123456@example.com" Remote-Address="192.0.2.0" Remote-Port="53991"
```

```
Name="administrator.configuration" Message="Configuration changed" User="admin"
Type="Virtual Meeting Room" ID="Alice's VMR" Fields="Description, Show names of
participants" Remote-Address="192.0.2.0" Remote-Port="53991"
```

administrator.conference

This module logs conference activity and break-in prevention policy triggers.

Parameters for messages related to conference activity

Each log entry includes information in the form of a series of name=value pairs. These are defined as follows:

Parameter	Definition
Call-id	An identifier that allows correlation of messages from the same call.
Conference	The name of the service being connected to.
ConferenceAlias	The alias that was dialed in order to connect to the service.
Conversation-id	An identifier that allows correlation of messages across separate "calls" for video+audio, RDP, chat for Skype for Business / Lync connections.
Detail	The reason the call was disconnected (for disconnect events). For more information, see Disconnection reasons .
Direction	Either: <ul style="list-style-type: none"> in: a call into Pexip Infinity out: a call dialed out from Pexip Infinity.
DisplayName	The display name of the participant.
Duration	The duration of the call (for disconnect events).
License-type	The type of license used in the call ("port", "audio" or "None").
Licenses	The number of licenses allocated for the call.
Location	The system location of the Conferencing Node that is handling the media processing for the call.
Media-node	The IP address of the Conferencing Node handling the media processing for the call.
Participant	The name or alias of the conference participant.
Participant-id	Allows correlation of messages from the same participant URI per connection type, for example an API connection will have a different Participant-id from a WebRTC connection.
Protocol	The protocol of the call (may also indicate "BACKPLANE" when a backplane between Conferencing Nodes is established or removed).
Proxy-location	The system location of the Proxying Edge Node.
Proxy-node	The IP address of the Proxying Edge Node.
Remote-address	The source IP address for the signaling for this call.
Requester	The Call ID of the participant requesting this operation.
Role	The role of this participant: <ul style="list-style-type: none"> chair: the participant is a Host guest: the participant is a Guest
Service-tag	A unique identifier used to track usage of this service.

Parameter	Definition
Service-type	The service type of the conference. This is one of the following: <ul style="list-style-type: none"> • conference: a Virtual Meeting Room • lecture : a Virtual Auditorium • two_stage_dialing: a Virtual Reception • test_call: a Test Call Service call • gateway: an Infinity Gateway call.
Signaling-location	The name of the Pexip Infinity location handling the signaling for this call.
Signaling-node	The IP address of the Conferencing Node handling the signaling for the call.
Vendor	System details about the endpoint of the participant, such as manufacturer name and version number for hard endpoints, or browser and operating system details for soft clients.

Examples

Trigger for the log message	Example log entry
A participant tries to connect to a conference (at this stage it is the conference alias that is logged).	<pre>Message="Participant attempting to join conference." ConferenceAlias="sip:meet.alice@example.com" Participant="sip:alice@example.com" Protocol="SIP" Direction="in" Remote-address="10.47.2.55" Participant-id="44be63a0-...c52" Registered="False" Location="Europe"</pre>
A conference instance is created on a Conferencing Node. This occurs when the first participant attempts to join this conference on that Conferencing Node. The Conference parameter indicates the name of the service, not the alias being dialed.	<pre>Message="Conference has been created." Conference="Meet Alice" Service-tag="" Service- type="conference" Conference-ID="cc7f6255-0986-447d-8ee9-561c5bbf4a24"</pre>
A participant joins the conference. This log message is issued before a role is assigned.	<pre>Message="Participant has joined." Conference="Meet Alice" Service-tag="" Service- type="conference" ConferenceAlias="sip:meet.alice@example.com" Participant="sip:alice@example.com" DisplayName="Alice Smith" Protocol="SIP" Direction="in" Vendor="TANDBERG/257 (TE4.1.1.273710)" Call-id="fe7107ba-...16c" Conversation- id="fe7107ba-...16c" Participant-id="44be63a0-...c52" Remote-address="10.47.2.55" Location="London" Licenses="1" Signaling-node="192.168.0.1" Signaling-location="London" Media- node="192.168.0.2" Conference-ID="cc7f6255-0986-447d-8ee9-561c5bbf4a24" Proxy-node="10.47.2.46" Proxy-location="London Proxy"</pre>
Conference creation fails.	<pre>Message="Failed creating conference." Conference="meet" Service-tag=""</pre>
A participant cannot join the conference. The Reason may vary as appropriate.	<pre>Message="Participant failed to join conference." ConferenceAlias="sip:meet.invalid@example.com" Participant="sip:bob@example.com" Protocol="SIP" Direction="in" Remote-address="10.44.100.75" Participant-id="6696b7fe-...2cb" Reason="Neither conference nor gateway found"</pre>
If the reason is "No direct route between Edge and Transcoding", the most likely cause is that local or external media policy is in place and it has nominated a Transcoding Conferencing Node that cannot be directly reached from the Proxying Edge Node that received the call.	<p>other reasons include:</p> <pre>Reason="No direct route between Edge and Transcoding" Reason="Out of proxying resource" Reason="Out of transcoding resource" Reason="System in maintenance mode"</pre>

Trigger for the log message	Example log entry
A participant leaves the conference. Includes the Duration (in seconds) for which they were present in the conference. This message provides the definitive record of the length of time a call was connected to a conference.	<pre> Message="Participant has disconnected." Conference="Meet Alice" Service-tag="" Service- type="conference" ConferenceAlias="sip:meet.alice@example.com" Participant="sip:alice@example.com" DisplayName="Alice Smith" Protocol="SIP" Direction="in" Call-id="fe7107ba-...16c" Conversation-id="fe7107ba-...16c" Participant-id="44be63a0-...c52" Remote-address="10.47.2.237" Location="London" Licenses="1" License-type="port" Signaling- node="192.168.0.1" Signaling-location="London" Media-node="192.168.0.1" Conference-ID="cc7f6255- 0986-447d-8ee9-561c5bbf4a24" Proxy-node="10.47.2.46" Proxy-location="London Proxy" Duration="63.123" Detail="Remote disconnect" </pre>
For more information on the reasons given in the Detail field, see Disconnection reasons .	
A conference stops on a particular Conferencing Node. The Duration of this conference is given in seconds.	<pre> Message="Conference has been stopped." Conference="Meet Alice" Service-tag="" Service- type="conference" Duration="62.234" Conference-ID="cc7f6255-0986-447d-8ee9-561c5bbf4a24" </pre>
A participant starts/stops presenting.	<pre> Message="Participant is presenting." Conference="meet" Service-tag="" Participant="sip:chuck@example.com" Call-id="e5dbd44b-...243" Conversation-id="e5dbd44b-...243" Message="Participant has stopped presenting." Conference="meet" Service-tag="" Participant="sip:chuck@example.com" Call-id="e5dbd44b-...243" Conversation-id="e5dbd44b-...243" </pre>
A participant using an Infinity Connect client, or any third party using the Pexip Client API, controls participants in a conference.	<pre> Message="Participant hold requested by API." Conference="meet" Service-tag="" Participant="sip:alice@example.com" Call-id="e5dbd44b-...243" Conversation-id="e5dbd44b-...243" Message="Participant resume requested by API." Conference="meet" Service-tag="" Participant="sip:alice@example.com" Call-id="e5dbd44b-...243" Conversation-id="e5dbd44b-...243" Message="Participant disconnect requested by API." Conference="meet.alice" Service-tag="" Requester="76cdf154-...0ce" Participant="sip:bob@example.com" Call-id="1aba49d2-...@10.44.26.36" Conversation-id="284b7a3d-...68b" Message="Participant mute requested by API." Conference="meet.alice" Service-tag="" Requester="b21e28cf-...d29" Participant="sip:bob@example.com" Call-id="fff88ae8-...@10.44.26.36" Conversation-id="9157b68a-...230" Message="Participant unmute requested by API." Conference="meet.alice" Service-tag="" Requester="b21e28cf-...d29" Participant="sip:bob@example.com" Call-id="fff88ae8-...@10.44.26.36" Conversation-id="9157b68a-...230" Message="Participant presentation receipt enabled by API." Conference="meet" Service-tag="" Participant="sip:alice@example.com" Call-id="e5dbd44b-...243" Conversation-id="e5dbd44b-...243" Message="Participant presentation receipt disabled by API." Conference="meet" Service-tag="" Participant="sip:alice@example.com" Call-id="e5dbd44b-...243" Conversation-id="e5dbd44b-...243" Message="Outgoing call requested" Conference="meet" Participant="sip:carol@example.com" Protocol="SIP" Role="chair" Message="Sending DTMF to participant." Conference="meet" Service-tag="" Participant="sip:alice@example.com" Call-id="e5dbd44b-...243" Conversation-id="e5dbd44b-...243" Digits="4" Message="Participant transfer requested by API." Conference="meet.alice" Service-tag="" Requester="66eb0873-...8ce" Participant="sip:bob@example.com" Call-id="d583e3a1-...@10.44.26.36" Conversation-id="2a8ea679-...8a8" To-Conference="3259oknfaoi" </pre>
A participant uses an Infinity Connect client or DTMF commands to control the conference.	<pre> Message="Conference terminated by participant" Conference="meet.alice" Service-tag="" Call- id="11d816fe-...ac4" Message="Conference lock requested by participant" Conference="meet.alice" Service-tag="" Call- id="221b63c69-...636" Message="Conference unlock requested by participant" Conference="meet.alice" Service-tag="" Call-id="221b63c69-...636" Message="Mute all guests requested by participant" Conference="meet.alice" Service-tag="" Call- id="221b63c69-...636" Message="Unmute all guests requested by participant" Conference="meet.alice" Service-tag="" Call-id="221b63c69-...636" </pre>

Trigger for the log message	Example log entry
Participants attempt to join a PIN-protected conference.	<pre> Message="PIN entry correct." Participant="sip:bob@example.com" ConferenceAlias="sip:meet@example.com" Call-id="27efdd36...6be" Role="host" Detail="Participant entered correct conference PIN and will be placed into the conference." Message="Participant entered incorrect conference PIN" RetriesRemaining="2" Participant="sip:bob@example.com" ConferenceAlias="sip:meet@example.com" Call- id="27efdd36...6be" Message="Participant exceeded PIN entry retries. Call will be disconnected." Participant="sip:bob@example.com" ConferenceAlias="sip:meet@example.com" Call- id="27efdd36...6be" Message="PIN entry timed out." Participant="sip:bob@example.com" ConferenceAlias="sip:meet@example.com" Call-id="a15dc445-...dfc" Detail="The correct PIN has not been received. Call will be disconnected." </pre>
Participants are involved in a PIN-protected conference.	<pre> Message="Host participant joining." Participant="sip:alice@example.com" ConferenceAlias="sip:meet@example.com" Call-id="4d3ad9a1-...833" Role="chair" Detail="Participant will be automatically placed into the conference." Message="Conference host timeout." Participant="sip:bob@example.com" ConferenceAlias="sip:meet@example.com" Call-id="4d3ad9a1-...833" Detail="The conference host has not joined. Call will be disconnected." Message="Guest participant waiting." Participant="sip:bob@example.com" ConferenceAlias="sip:meet@example.com" Call-id="50a86594-...797" Role="guest" Detail="Waiting until participant is allowed to join the conference." Message="Guest participant joining." Participant="sip:bob@example.com" ConferenceAlias="sip:meet@example.com" Call-id="50a86594-...797" Role="guest" Detail="Participant will be automatically placed into the conference." </pre>
A conference is automatically ended.	<pre> Message="Last host left the conference and has not re-joined; conference will be terminated." Conference="Meet Alice" Service-tag="" Message="Removing participant due to conference termination" Participant="sip:bob@example.com" Conference="Meet Alice" Service-tag="" </pre>

Break-in prevention policy example log messages

The following examples show messages that may be logged by the [break-in prevention policies](#).

Logged when PIN brute force resistance has temporarily disabled a service, and for all subsequent attempts while the service is blocked:

```

Message="Break-in prevention policy blocking all attempts to join this service." ConferenceAlias="alice" Service="Alice's VMR"
Participant="Crooky McCrookface" Protocol="API" Direction="in" Remote-address="10.44.21.35" Reason="Service appears to be under PIN
break-in attack" remaining_block_duration_seconds="525"

```

Logged when VOIP scanner resistance has temporarily blocked an address:

```

Message="Participant has been quarantined by Break-in prevention policy due to excessive failed join attempts." Participant="Crooky
McCrookface" Protocol="API" Direction="in" Remote-address="10.44.21.35" Reason="Too many attempts to join non-existent aliases"
remaining_block_duration_seconds="488"

```

and then any subsequent attempts generate messages such as:

```

Message="Break-in prevention policy rejecting call attempt from quarantined caller." Protocol="API" Direction="in" Local-alias="
[u'alice']" Remote-address="10.47.250.169" Reason="Suspicious join attempt rejected" remaining_block_duration_seconds="519"

```

administrator.participantdialer

This module logs outbound calls placed from a conference or via the Infinity Gateway.

Examples

Logged when dialing out from a conference:

```

Message="Placing outbound call" ConferenceAlias="meet.alice@example.com" DisplayName="meet.alice" Participant-
alias="sip:bob@example.com" Protocol="SIP" Role="host"

```

Logged when making a call via the Infinity Gateway:

```

Message="Placing outbound call" Conference="Route to Lync:423a18c5-b06c-4654-9d34-e6b82f308b64" ConferenceAlias="alice@example.com"
DisplayName="Alice" Participant-alias="sip:carol@example.com" Protocol="MSSIP" Role="guest"

```

administrator.registration

This module logs endpoint registration activities.

Examples

Logged when a registration is added:

```
Message="Registration added" Alias="alice@example.com" Protocol="SIP" Registration-id="8695ea23-66fb-41d2-9266-d16598f5d133"
Natted="False" Location="Europe"
```

Logged when a registration is removed:

```
Message="Registration deleted" Alias="bob@example.com" Protocol="WebRTC" Registration-id="6936f540-6dbc-40ac-9d55-5469bbf4e561"
Location="Europe" Reason="Registration expired"
```

administrator.ldap.sync

This module logs VMR, device and user synchronization template activities.

Examples

Logged when a VMR template synchronization is initiated:

```
Message="Beginning VMR Sync" Template="All employees" Creating="219" Deleting="0" Modifying="0" Unchanged="0"
```

Logged when a VMR template synchronization finishes:

```
Message="Completed VMR Sync" Template="Europe" Status="sync_succeeded" Created="18" Deleted="10" Updated="5" Unchanged="34"
Warnings="3" Last_warning="Conference clash: not overwriting manually created conference"
```

Logged when an alias clash occurs during VMR template synchronization:

```
Message="Sync detected alias clash: alias held by an existing conference has not been assigned to the newly synced conference."
NewConference="Alice's VMR" ExistingConference="Bob's VMR" Alias="meet@example.com"
```

administrator.apps.cloudbursting

This module logs cloud bursting activities.

Examples

Logged when an overflow Conferencing Node is started up:

```
Message="On-prem location 'Oslo' is overloaded! Starting instance i-0043088a9073b57e6 (stopped) in location 'AWS eu-west-1'"
```

Logged when an overflow Conferencing Node is shut down:

```
Message="Shutting down idle worker i-597394d2 (running) (uptime: 50 minutes)"
```

administrator.alarm

This module logs all Alarm raised and Alarm lowered events.

Examples

Logged when a "connectivity lost" alarm is lowered:

```
"Message="Alarm lowered" Node="10.47.2.46" Alarm="connectivity_lost" Instance="Source=10.47.2.46, Destination=10.47.2.17"
```

administrator.web

This module logs activity relating to conference and participant control by an administrator using the Administrator interface.

Examples

```
Message="Initiated mute command." User="admin" Participant-alias="Alice" Participant-
id="25fca17e-...374"
```

```
Message="Initiated unmute command." User="admin" Participant-alias="Alice" Participant-
id="25fca17e-...374"
```

```
Message="Initiated transfer command." User="admin" Participant-alias="Alice" Participant-
id="25fca17e-...374" Conference-alias="meet.alice2" Role="chair"
```

```
Message="Initiated conference lock." User="admin" Conference="meet.alice2"
```

```
Message="Initiated conference unlock." User="admin" Conference="meet.alice2"
```

```
Message="Initiated disconnect command." User="admin" Participant-alias="Alice" Participant-id="21300b40-...557"
```

```
Message="Initiated conference disconnect." User="admin" Conference="meet.alice"
```

administrator.rtmp

This module logs activity related to connections over RTMP.

Example

Logged when someone tries to dial in to a mandatory encrypted conference using RTMP but the SIP TLS FQDN is not configured and so Pexip Infinity cannot do RTMPS:

```
Message="SIP TLS FQDN must be configured before encrypted RTMP calls can be accepted"
```

administrator.scheduling

This module logs the activities of the VMR Scheduling for Exchange service.

Parameters

Parameter	Definition
ExchangeConnectorID	The database ID of the Pexip Exchange Integration which is the cause of this log activity.
CorrelationID	A UUID which identifies which specific item this log refers to. The logs can be searched for an item's CorrelationID to easily find all logs pertaining to that item.
Name	The name of the VMR associated with the scheduled conference.
VmsID	The database ID of the VMR associated with the scheduled conference.
ScheduledAliasID	The database ID of the scheduled alias associated with the scheduled conference.
UUID	The UUID which identifies the Alias associated with the scheduled conference (this is the UUID used for the PXPS tag).
Alias	The full alias of the VMR associated with the scheduled conference, including the domain part
NumericAlias	The alias of the VMR associated with the scheduled conference, without the domain part
ScheduledConferenceID	The database ID of the single conference.
RecurringConferenceID	The database ID of the recurring conference.
OccurrenceScheduledConferenceIDs	The database IDs of the individual conference occurrences associated with the recurring conference. This is a list of values separated by commas.

Examples

Trigger for the log message	Example
New scheduled alias created.	Message="Scheduled Alias added" ScheduledAliasID="4983" UUID="5b195574-d8e0-4f5a-b07e-7826f419db7f" Alias="8359689@rd.pexip.com" NumericAlias="8359689" ExchangeConnectorID="2"
New single meeting created.	Message="Scheduled Conference added" VmsID="35111" Name="Example new meeting (Toby Finch)" ScheduledConferenceID="5001" ScheduledAliasID="4983" ExchangeConnectorID="2" CorrelationID="4d566414-ald3-5c3f-aa08-67fda6737fa7"
Recurring meeting changed to a single meeting.	Message="Recurring Conference changed to Scheduled Conference" VmsID="35000" Name="Example meeting (Jim Bob)" RecurringConferenceID="1" ScheduledConferenceID="10" ExchangeConnectorID="1" CorrelationID="e55af8b7-ec66-4376-8521-16a41de918eb"
The <code>RecurringConferenceID</code> is deleted and the new <code>ScheduledConferenceID</code> assigned.	

Trigger for the log message	Example
Single meeting updated.	Message="Scheduled Conference changed" VmsID="35111" Name="Example edited meeting (Toby Finch)" ScheduledConferenceID="5001" ExchangeConnectorID="2" CorrelationID="4d566414-ald3-5c3f-aa08-67fda6737fa7"
New recurring meeting created.	Message="Recurring Conference added" VmsID="1234" Name="Example new recurring meeting (John Smith)" RecurringConferenceID="1000" OccurrenceScheduledConferenceIDs="501, 502" ScheduledAliasID="2000" ExchangeConnectorID="1" CorrelationID="5a304f92-6705-4f45-87b3-428d8a7261ef"
Single meeting changed to a recurring meeting. The <code>ScheduledConferenceID</code> is deleted and the new <code>RecurringConferenceID</code> assigned.	Message="Scheduled Conference changed to Recurring Conference" VmsID="10" Name="Another example meeting (Darth Vader)" ScheduledConferenceID="12" RecurringConferenceID="14" OccurrenceScheduledConferenceIDs="13, 14, 15" ExchangeConnectorID="1" CorrelationID="ab40e1f4-5a46-494a-a39f-8156ded538e"
Recurring meeting updated. The <code>OccurrenceScheduledConferenceIDs</code> replace any occurrences which already exist.	Message="Recurring Conference changed" VmsID="1234" Name="Example edited recurring meeting (John Smith)" RecurringConferenceID="1000" OccurrenceScheduledConferenceIDs="503" ExchangeConnectorID="1" CorrelationID="5a304f92-6705-4f45-87b3-428d8a7261ef"
An individual occurrence of an existing recurring conference is updated but the occurrence is not currently in the database. Modified occurrences are always added to the database.	Message="Occurrence Scheduled Conference added" VmsID="1234" Name="Example edited recurring meeting (John Smith)" OccurrenceScheduledConferenceID="504" RecurringConferenceID="1000" ExchangeConnectorID="1" CorrelationID="5a304f92-6705-4f45-87b3-428d8a7261ef"
An individual occurrence of an existing recurring conference is updated and the occurrence is already in the database.	Message="Occurrence Scheduled Conference changed" VmsID="1234" Name="Example edited recurring meeting (John Smith)" OccurrenceScheduledConferenceID="504" ExchangeConnectorID="1" CorrelationID="5a304f92-6705-4f45-87b3-428d8a7261ef"
Meeting canceled — this normally happens when a single meeting or entire recurring meeting is canceled, but it may also happen if an item is rejected because an update to it was invalid. Note that deleting the conference also causes the associated single conference or recurring conference and any related aliases to be deleted at the same time.	Message="Conference deleted" VmsID="35111" ExchangeConnectorID="2" CorrelationID="4d566414-ald3-5c3f-aa08-67fda6737fa7"
Occurrence canceled — this normally happens when an individual occurrence in a recurring series is canceled, but it may also happen if the occurrence is rejected because an update to it was invalid. Note that the <code>VmsID</code> is left unchanged.	Message="Occurrence Scheduled Conference deleted" VmsID="35111" OccurrenceScheduledConferenceID="504" ExchangeConnectorID="1" CorrelationID="5a304f92-6705-4f45-87b3-428d8a7261ef"
Single meeting recovered while running the Scheduling Recovery tool. This means a single meeting was added back to the database with a new alias.	Message="Scheduled Conference recovered" VmsID="10" Name="Example recovered meeting (Fred)" ScheduledConferenceID="12" ScheduledAliasID="12" ExchangeConnectorID="1" CorrelationID="1b770df6-d27f-4c32-b499-1e3104c2f45b"
Recurring meeting recovered while running the Scheduling Recovery tool. This means a recurring meeting was added back to the database with a new alias.	Message="Recurring Conference recovered" VmsID="2" Name="Example recovered recurring meeting" RecurringConferenceID="2" OccurrenceScheduledConferenceIDs="2, 3" ScheduledAliasID="3" ExchangeConnectorID="2" CorrelationID="65dc5d19-c8fb-4040-9dd5-447ef196eda2"

Trigger for the log message	Example
<p>The scheduling background tasks successfully updated a recurring conference.</p> <p><code>NewOccurrenceScheduledConferenceIDs</code> can be empty if no new occurrences were added at this point.</p> <p><code>IsDepleted</code> indicates whether there are any more occurrences left in this recurring series.</p>	<pre>Message="Background tasks updated Recurring Conference" VmsID="42" Name="Meaning of life" RecurringConferenceID="42" NewOccurrenceScheduledConferenceIDs="" CurrentIndex="3" IsDepleted="False" ExchangeConnectorID="1" CorrelationID="79e4ee44-3a05-41b2-b988-4e6d98c15072"</pre>
<p>The scheduling background tasks encountered an error when updating a recurring conference.</p> <p>This can happen if we can no longer update a recurring series. For example, the room/equipment resource could have been changed and this was a recurring series stored on the old resource.</p>	<pre>Message="Recurring Conference deleted because it can no longer be processed" VmsID="13" RecurringConferenceID="13" ExchangeConnectorID="2" CorrelationID="e86b9fca-c0ba-4ef2-8b56-8f75fcb60d12"</pre>
<p>The scheduling background tasks successfully deleted an expired single meeting.</p>	<pre>Message="Background tasks deleted expired Scheduled Conference" VmsID="101" ScheduledConferenceID="201" ExchangeConnectorID="1" CorrelationID="359803da-f34f-4515-b505-313b3c9f53fc"</pre>
<p>The scheduling background tasks successfully deleted an expired meeting occurrence.</p> <p>Note that when an individual occurrence expires, it is only the corresponding <code>OccurrenceScheduledConferenceID</code> that gets deleted. There may still be other occurrences which are not expired.</p>	<pre>Message="Background tasks deleted expired Occurrence Scheduled Conference" VmsID="14" OccurrenceScheduledConferenceID="15" ExchangeConnectorID="1" CorrelationID="375c005e-5edc-4a44-876c-43c8cedf6657"</pre>
<p>The scheduling background tasks successfully deleted an expired recurring conference.</p>	<pre>Message="Background tasks deleted expired Recurring Conference" VmsID="53" RecurringConferenceID="33" ExchangeConnectorID="3" CorrelationID="a874129f-da34-4bfb-8422-211711af52d0"</pre>
<p>The scheduling background tasks successfully deleted a used scheduled alias.</p>	<pre>Message="Background tasks deleted expired used Scheduled Alias" ScheduledAliasID="5" ExchangeConnectorID="2"</pre>
<p>The scheduling background tasks successfully deleted an unused scheduled alias.</p>	<pre>Message="Background tasks deleted expired unused Scheduled Alias" ScheduledAliasID="7" ExchangeConnectorID="2"</pre>

administrator.otj

This module logs the activities of the One-Touch Join service when:

- a One-Touch Join meeting is created
- a One-Touch Join meeting is deleted
- a One-Touch Join meeting is changed.

Parameters

Parameter	Definition
OTJProfileName	The name of the OTJ Profile associated with this meeting.
OTJProfileID	The ID of this OTJ Profile.
Room	The email address of the room resource in whose calendar the meeting has been scheduled.
Subject	The text that appears in the subject line of the meeting invitation.
OrganizerEmail	The email address of the person who created the meeting invitation.
StartTime	The scheduled start time of the meeting.
EndTime	The scheduled end time of the meeting.
Alias	The alias that the endpoint will use to dial in to the meeting.
OTJRuleName	The name of the meeting processing rule that was matched and used to process this meeting.

Examples

Trigger for the log message	Example
New OTJ meeting created	Message="OTJ Meeting Created" OTJProfileName="Test MJX integration0" OTJProfileID="1" Room="roomresource@resource.calendar.google.com" Subject="Test Meeting" OrganizerEmail="jack@pexip.com" StartTime="2020-02-05 15:00:00" EndTime="2020-02-05 16:00:00" Alias="1234567890@pexip.com" OTJRuleName="Pexip Rule"
Existing OTJ meeting changed	Message="OTJ Meeting Changed" OTJProfileName="Test MJX integration0" OTJProfileID="1" Room="roomresource@resource.calendar.google.com" Subject="Test Meeting" OrganizerEmail="jack@pexip.com" StartTime="2020-02-05 15:30:00" EndTime="2020-02-05 16:30:00" Alias="1234567890@pexip.com" OTJRuleName="Pexip Rule"
Existing OTJ meeting deleted	Message="OTJ Meeting Deleted" OTJProfileName="Test MJX integration0" OTJProfileID="1" Room="roomresource@resource.calendar.google.com" Subject="Test Meeting" OrganizerEmail="jack@pexip.com" StartTime="2020-02-05 15:30:00" EndTime="2020-02-05 16:30:00" Alias="1234567890@pexip.com" OTJRuleName="Pexip Rule"

Support log system modules

Support logs provide a more detailed record of activity on the Pexip Infinity system:

Module	Contains
support.bfcp	BFCP signaling.
support.conference	Logs the aliases that are matched against Call Routing Rules. The parsed or unparsed alias match type depends upon the rule's Match against full alias URI setting. For example: Message="Alias matched gateway rule" Rule="Route calls from Lync" Description="" Service-tag="" Unparsed-alias="alice@example.com" Parsed-alias="alice@example.com" Matched-alias-type="Parsed-alias"
support.conferenceservice	Logs when a participant has sent a chat message (but does not include the message contents).
support.dns	DNS lookups associated with SIP signaling.
support.event	Errors when attempting to connect to an external event sink server.
support.externalpolicy	Requests sent to an external policy server and the associated response.
support.gms	Gateway calls into Google Meet.

Module	Contains
support.h323.q931, support.h323.ras, support.h323.h245	H.323 signaling.
support.jinja2	Messages generated by the pex_debug_log filter when inserted into local policy jinja2 scripts.
support.otj	Messages related to One-Touch Join.
support.participant	<p>Details about conference participants, including presentation stream activation, loss of incoming video, and call media statistics when a call is completed.</p> <p>An example of end-of-call media statistics is as follows:</p> <pre> Message="Media Stream destroyed" Participant="sip:alice@example.com" Call- id="890b38520c27737c@192.168.0.1" Conversation-id="1238035a...a2de" Detail="Stream 0 (audio) RX: rate 54kbps loss 0.77% jitter 6ms TX: rate 61kbps loss 0.00% jitter 2ms codec MP4A-LATM rate 64kbps" Message="Media Stream destroyed" Participant="sip:alice@example.com" Call- id="890b38520c27737c@192.168.0.1" Conversation-id="1238035a...a2de" Detail="Stream 1 (video) RX: rate 1085kbps loss 0.00% jitter 3ms TX: rate 699kbps loss 0.00% jitter 5ms codec H264 rate 700kbps resolution 768x448 fps 30" Message="Current participant is the presenter" Participant="sip:bob@example.com" Call- id="c15b773a81af1740@192.168.0.1" Conversation-id="1238035a...a2de" Message="Presentation started" Presenter="sip:bob@example.com" Participant="sip:alice@example.com" Call- id="890b38520c27737c@192.168.0.1" Conversation-id="1238035a...a2de" Message="Stopped sending presentation to participant" Last-presenter="sip:bob@example.com" Participant="sip:alice@example.com" Call-id=890b38520c27737c@192.168.0.1" Conversation- id="1238035a...a2de" Message="Call is on hold" Participant="sip:alice@example.com" Call-id="890b38520c27737c@192.168.0.1" Conversation-id="1238035a...a2de" Message="Call has resumed" Participant="sip:alice@example.com" Call-id="890b38520c27737c@192.168.0.1" Conversation-id="1238035a...a2de" Message="Lost incoming video" Participant="sip:alice@example.com" Call-id="890b38520c27737c@192.168.0.1" Conversation-id="1238035a...a2de" </pre>
support.pulse	<p>Logs scheduled maintenance events in Azure. For example:</p> <pre> New scheduled maintenance event: {'Description': 'The system will Reboot', 'EventId': '2262d851-75e0-11eb- 92d9-000d3abdf5e0', 'EventSource': 'Platform', 'EventStatus': 'Scheduled', 'EventType': 'Reboot', 'NotBefore': '2021-01-06T11:10:18+00:00', 'ResourceType': 'VirtualMachine', 'Resources': ['azure-mcu001']} </pre>
support.rest	Messages to and from the Pexip Client API (used by the Infinity Connect mobile client, Infinity Connect client, and other third parties).
support.rtmp	Calls (streaming) to an RTMP device.
support.scheduled_ maintenance_events	<p>Logs actions taken by Pexip Infinity resulting from detecting scheduled maintenance events in Azure. For example:</p> <pre> Message="Scheduled maintenance event affecting MCU node has been detected. System will be placed into maintenance mode." Message="Maintenance event completed. Maintenance mode will be unset." </pre>
support.sip	SIP signaling.
support.teams	Gateway calls into Microsoft Teams.

Content of signaling messages

All log lines for signaling-related messages contain source and destination IP addresses and ports.

Multi-line messages are encoded on a single line with the character sequence `^M` representing a linebreak.

In addition to observing remote IP addresses, for H323 logging, a `Uuid` is present for H.225 (RAS) and H.245 signaling messages. These come from the same IP address but different ports, and the Uuid allows you to identify that they relate to the same call.

For BFCP signaling, all log lines are tied to a SIP `Call-id`.

Most of the content of the support log is the signaling messages themselves; however it also includes logging such as TCP and TLS connection attempts, successes, and failures.

Creating and viewing diagnostic graphs

The Pexip Infinity Administrator interface includes a series of graphs that can be used to monitor the status and performance of the platform. You can view and edit the default graphs, create your own graphs, and change the order in which the various graphs appear.

To view and manage the graphs, go to **Status > Diagnostic Graphs**.

Information shown in the graphs

The graphs show data for the selected metrics starting from either when the relevant node was created, or when it was first rebooted following a platform upgrade to version 20 or later. Further metrics may be added in subsequent releases of Pexip Infinity; when you upgrade a node, data for any newly-added metrics will be available starting from the point at which the node was rebooted following the upgrade.

If a node has been temporarily out of service at any point, there will be a corresponding gap in the graphs. This also applies to cloud bursting nodes for the times that they were not in use.

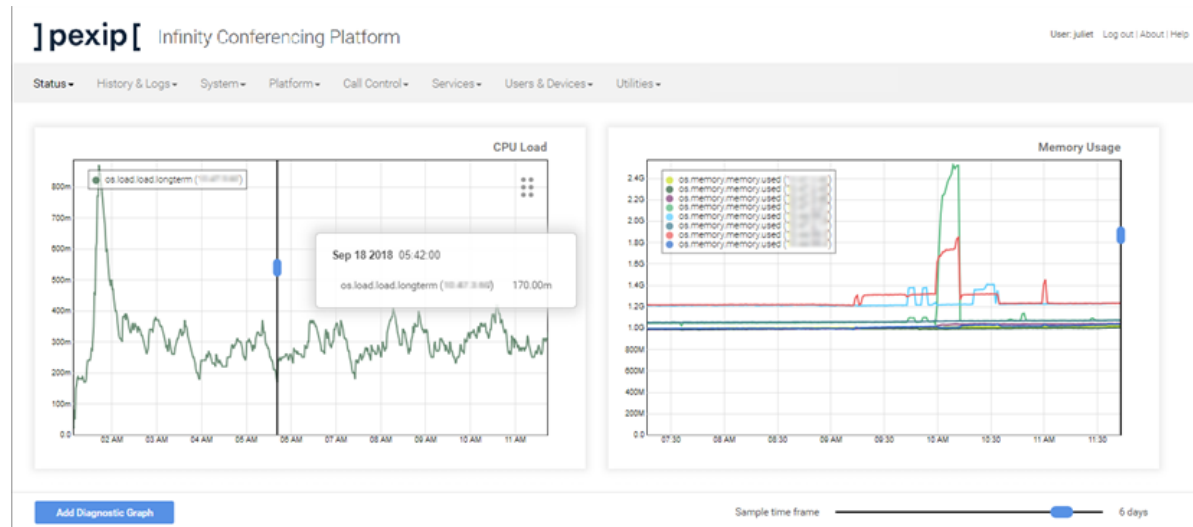
Diagnostic graphs are generated dynamically from information obtained from the relevant nodes, so the node must be currently available in order for its measurements to be shown in the graph. This means that any nodes that are currently out of service are not shown; nor are any cloud bursting nodes, unless they are currently in use.

Default graphs


Two graphs are included by default:

- **Memory usage** shows the memory being used by the Management Node's application and OS processes
- **CPU load** shows the system load on the Management Node.

Viewing and controlling graphs



- To change the time period covered by all graphs shown on the page, drag the **Sample time frame** control at the bottom right of the window to the left to zoom out, or right to zoom in.
- To change the time period covered by an individual graph, select the blue handle on the vertical bar at the right of the graph and drag it downwards to zoom out, or upwards to zoom in. Note that in some cases after zooming in or out the lines may be cropped. To fix this, click on the line twice.
- To view the values at a particular point in the graph, drag the vertical bar to the point in time you're interested in. The values at that point will appear in a pop-up next to the control.
- To toggle individual lines of the graph on or off, click on them from within the legend. Those that are available but not currently showing will appear grayed out.


- To move a graph, select the  icon at the top right of the graph, and drag it to the desired location.

Creating new graphs

You can create new custom graphs to add to the **Diagnostic Graphs** page. Newly-created graphs will still show retrospective information, starting from when the selected node was upgraded to version 20 or later.

1. From the bottom left of the window, select **Add Diagnostic Graph**.
2. In the field above the new graph, enter an appropriate title.
3. From the **Available nodes** and **Available metrics** boxes below the graph, select the combination(s) you wish to appear in the graph.
 - For more information on a particular metric, hover over it — the description will appear in a tooltip.
 - To select multiple items, hold down the **Ctrl** button (Windows) or **Command** button (Mac).
 - The metrics that are available depend on whether the selected node is a Management Node, Transcoding Conferencing Node, or Proxying Edge Node.
 - If you have selected multiple nodes, only those metrics that are common to all selected node types are available.

When you have made your selection, select **Add selected metric**.

4. The chosen combinations will appear in the **Selected metrics** table. From here you can refine the appearance of the graph, such as changing the color of individual lines or changing the magnification.
 -  If a selected metric does not yet have a value (for example, if you have selected the count of node allocations and none have yet taken place), the **Minimum**, **Average**, **Maximum** and **Last** values will be blank.
5. From the bottom left of the window, select **Save**.

The new graph will be added to the **Diagnostic Graphs** page. Click on the graph to drag it to the desired location.

Editing and deleting graphs

To **edit** an existing graph, double-click it. You can then change the name, add and remove metrics, and change the appearance of the graph in the same way as when creating a new graph. When you have finished, from the bottom left of the window select **Save**.

To **delete** a graph, double-click it. From the bottom right of the window, select **Delete**.

Disconnection reasons

When a `Participant has disconnected` message appears in the [admin log](#), further information about the reason will be given in the `Details` field of the message.

This **Disconnect** reason is also shown when [viewing historical information about a participant](#).

The table below lists reasons for disconnections that may be given, along with an explanation of each. Note that this is not an exhaustive list, because in some cases the disconnect reason is provided by the endpoint.

Any **Abnormal** events are not usually seen during the expected functioning of the system and may require further investigation or intervention.

Reason for disconnection	Normal or Abnormal event	Message code	Meaning/resolution
Conference host ended the conference with a DTMF command	Normal	#pex120	A Host participant ended the call using a DTMF command.
Conference terminated by a Host participant	Normal	#pex121	An Infinity Connect Host participant has selected "disconnect all", or a client API command was used to terminate the conference.
Conference terminated by an administrator	Normal	#pex122	An administrator using the Pexip Infinity Administrator interface has selected "disconnect all", or a management API command was used to end the conference.
Disconnected by an administrator	Normal	#pex123	An administrator using the Pexip Infinity Administrator interface has disconnected this particular participant.
Disconnected by another participant	Normal	#pex124	A Host using an Infinity Connect client has disconnected a specific participant.
Conference terminated by another participant	Normal	#pex125	An Infinity Connect Host participant has selected "disconnect all", or a client API command was used to terminate the conference.
Timeout waiting for conference host to join or permit access to locked conference	Normal	#pex126	The participant timed out because the conference Host either did not join the conference, or did not permit the participant to join a locked conference.
Signaling node disconnected	Abnormal	#pex129	The media node lost connectivity to the signaling node.
Media process disconnected	Abnormal	#pex130	The Conferencing Node hosting the media has encountered an unexpected behavior.
Media node disconnected	Abnormal	#pex131	The signaling node lost connectivity to the media node.
Proxied participant disconnected	Abnormal	#pex132	The proxying node lost connectivity to the transcoding node.
No participants can keep conference alive	Normal	#pex140	This was the only remaining participant, and they were an ADP that was not configured to keep the conference alive.
All conference hosts departed hosted conference	Normal	#pex141	There are no Host participants remaining in the conference.
Last remaining participant removed from conference after timeout	Normal	#pex142	This was the only participant remaining, and they were disconnected after the configured amount of time.
Test call finished	Normal	#pex143	This was a call to the Test Call Service that was automatically disconnected after the specified time.
Call rejected	Normal	#pex150	The person being called did not answer or could not be reached.

Reason for disconnection	Normal or Abnormal event	Message code	Meaning/resolution
Call disconnected	Normal	#pex151	An Infinity Connect client has been disconnected by themselves or another system other than Pexip Infinity.
Call failed - please contact your administrator		#pex158	A call routing rule with an invalid regex replace string has been configured.
Failed to gather IP addresses.	Abnormal	#pex170	The browser cannot find the local IP address. This may be due to ad blockers. An Infinity Connect WebRTC client could not determine its IP address. This may be because there are privacy extensions installed.
Call Failed: Error: Could not get access to camera/microphone. Have you allowed access? Has any other application locked the camera?	Abnormal	#pex171	An Infinity Connect WebRTC participant has not allowed their camera or microphone to be shared, or has no camera or microphone available.
Timer expired awaiting token refresh	Abnormal	#pex190	An Infinity Connect WebRTC client was unable to refresh its token after 2 minutes. This is likely due to network issues.
Resource unavailable	Abnormal	#pex191	There was insufficient transcoding or proxying capacity on the Transcoding Conferencing Node or the Proxying Edge Node on which the call landed.
Participant exceeded PIN entry retries	Normal	#pex192	The participant exceeded the allowed number of PIN entry attempts (3).
AVMCU cannot connect to PIN-protected or locked conference	Normal		There are some limitations with merging and escalating Skype for Business / Lync meetings with PIN-protected Pexip Infinity conferences.
Backplane disconnected	Abnormal		A connection between two Conferencing Nodes was lost, and the participant was connected to one of the nodes.
Browser closed	Normal		An Infinity Connect participant closed their web app or desktop client.
Call transferred	Normal		The participant was transferred to another conference.
CCCP call disconnected	Normal		The call from Pexip Infinity to the SfB/Lync server has been disconnected by a system other than Pexip Infinity. The reasons include: <ul style="list-style-type: none"> • Conference Terminated – Organizer Ended Session: the normal meeting ended scenario (e.g. the organizer pressed "End Meeting"). • Participant Removed: the gatewayed participant (VTC/VMR) was removed from the meeting by the organizer. • Conference Terminated – Enterprise User Absent: all of the SfB/Lync clients have left the meeting and the gateway participants have been timed out. • Conference Terminated – Inactivity: the meeting timed out due to inactivity (no new users have joined in the last 24 hours). • Conference Terminated: the organizer has ended the meeting. • RFC3263 lookup failure: the call could not be connected because DNS lookup failed.

Reason for disconnection	Normal or Abnormal event	Message code	Meaning/resolution
Conference number entry attempts exceeded	Normal		A participant using a Virtual Reception has exceed the allowed number of attempts (3) to enter a conference number. Does not apply to Infinity Connect clients.
Connection to the other side was lost in a non-clean fashion	Normal		An H.323 participant was disconnected by themselves or another system (other than Pexip Infinity).
Connection was closed cleanly	Normal		An H.323 participant was disconnected by themselves, another system, or Pexip Infinity.
Detected AVMCU call failure	Abnormal		Call to the Skype for Business / Lync server failed.
Dialog has failed	Abnormal		(SIP) A reverse connection could not be established. This is likely due to a TCP connection issue.
Insufficient capacity	Abnormal		There was insufficient transcoding capacity on the Conferencing Nodes within the location the call landed on, and any configured media overflow locations, or a participant limit was reached.
Local disconnect (No response to Round Trip Delay Request)	Normal		Pexip Infinity disconnected the participant because their endpoint did not respond to the request sent by Pexip Infinity.
No conference number entered	Normal		The participant exceeded the allowed time at the Virtual Reception (120 seconds). Does not apply to Infinity Connect clients.
Participant did not connect in time	Normal		The outgoing call was not answered.
Participant failed to start audio channel			An ACK was not received from the far side. This may be because the far side never received, or failed to process, the 200 OK from Pexip Infinity, and therefore never set up media.
PIN entry timed out	Normal		The participant exceeded the allowed time at the PIN entry prompt. Does not apply to Infinity Connect clients.
Presentation stopped	Normal		A Skype for Business / Lync participant stopped presenting.
RFC3263 lookup failure	Abnormal		The call could not be connected because DNS lookup failed.
Re-INVITE timer has expired	Abnormal		Failed to establish a connection to the far end in order to send a SIP INVITE message.
Remote disconnect	Normal		A SIP/H.323 participant was disconnected by themselves or another system (other than Pexip Infinity).
Session renegotiation failed	Abnormal		A reverse connection could not be established. This is likely due to a TLS (certificate) issue.
Timeout expired waiting for ACK	Abnormal		SIP: no ACK message was received within the default timeout period. This often occurs due to issues with DNS (such as unresolvable DNS records, or resolving to the wrong host) as referenced in the Record-Route or Contact SIP signaling headers.
Transaction failed INVITE <random string of characters>	Abnormal		SIP: INVITE message was not responded to within the default timeout period.

Reason for disconnection	Normal or Abnormal event	Message code	Meaning/resolution
Transaction failed UPDATE <random string of characters>	Abnormal		SIP: UPDATE message was not responded to within the default SIP timeout period (32 seconds).
Undefined reason	Normal		An H.323 participant has been disconnected by themselves or another system (other than Pexip Infinity).
User initiated disconnect	Normal		An Infinity Connect WebRTC participant manually disconnected themselves.

Pexip Infinity port usage and firewall guidance

The diagrams and tables below show the ports used when the Management Node and Conferencing Nodes connect to other devices.

Firewall, routing and NAT guidance

Note that in all Pexip Infinity deployment scenarios:

- The Management Node must be able to reach all Conferencing Nodes (Proxying Edge Nodes and Transcoding Conferencing Nodes) and vice versa.
- Each Conferencing Node must be able to reach every other Conferencing Node (Proxying Edge Nodes and Transcoding Conferencing Nodes), except:
 - When a location contains Proxying Edge Nodes, those nodes only require IPsec connectivity with:
 - any other proxying nodes in that location
 - all nodes in the transcoding location, and the primary and secondary overflow locations that are associated with that location
 - the Management Node.

This means that the proxying nodes in one location do not need to have a direct network connection to other proxying nodes in other locations.

- Any internal firewalls must be configured to allow UDP port 500 and traffic using IP protocol 50 (ESP) in both directions between all Pexip nodes.
- There cannot be a NAT between any Pexip nodes.

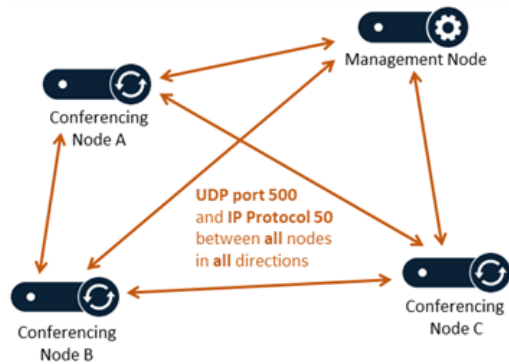
When a secondary network address is configured on a Conferencing Node:

- The primary address is always used for inter-node communication to the Management Node and to other Conferencing Nodes.
- SSH connections can be made only to the primary interface.
- The secondary address is always used for signaling and media (to endpoints and other video devices).
- Connections to DNS, SNMP, NTP, syslog and so on, go out from whichever interface is appropriate, based on routing.
- You can have a mixture of any number of single-interfaced and dual-interfaced Conferencing Nodes, providing all nodes can communicate with each other via their primary interfaces.

Inter-node communication (Conferencing Nodes and Management Node)

These are the port usage rules for all inter-node communication (local and remote) — between Conferencing Nodes, and between the Management Node and Conferencing Nodes:

Source address	Source port	Destination address	Dest. port	Protocol	Notes
Management Node	500	Conferencing Node	500	UDP	ISAKMP (IPsec) inter-node communication
Management Node	n/a	Conferencing Node	n/a	ESP	IPsec / IP Protocol 50 inter-node communication
Conferencing Node	500	Management Node / Conferencing Node	500	UDP	ISAKMP (IPsec) inter-node communication
Conferencing Node	n/a	Management Node / Conferencing Node	n/a	ESP	IPsec / IP Protocol 50 inter-node communication



Port requirements for inter-node communication

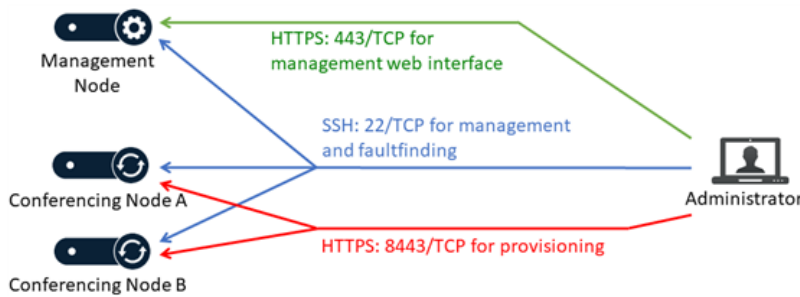
- **No NAT** between nodes – path must be directly routable
- **UDP port 500 (IKE)**, and **IP Protocol 50 (IPsec ESP)** to pass in **both directions**.
- If a location only contains Proxying Edge Nodes, then those proxying nodes in that location only require IPsec connectivity with any other proxying nodes in that location, the transcoding location, and the primary and secondary overflow locations, and with the Management Node.

Administration access

These are the port usage rules for administrative access to the Management Node and Conferencing Nodes:

Source address	Source port	Destination address	Dest. port	Protocol	Notes
SSH client	<any>	Management Node / Conferencing Node	22	TCP	SSH *
Web browser / API workstation	<any>	Management Node	80*/443	TCP (HTTP/HTTPS)	Management web and API administration
Web browser / API workstation	<any>	Conferencing Node	8443	TCP (HTTPS)	Provisioning a Conferencing Node (primarily for Azure/GCP/AWS deployments)

* Only required if you want to allow administrative access via this port.



Port requirements for administrative access

Peripheral services

These are the port usage rules for the mandatory and optional peripheral services used by the Management Node and Conferencing Nodes:

Source address	Source port	Destination address	Dest. port	Protocol	Notes
Standard features					

Source address	Source port	Destination address	Dest. port	Protocol	Notes
Management Node / Conferencing Node	55000–65535	DNS server	53	TCP/UDP	DNS
Management Node / Conferencing Node	123, 55000–65535	NTP server	123	UDP	NTP
Management Node	55000–65535	Pexip Licensing server (pexip.flexnetoperations.com and activation.pexip.com ◊)	443	TCP(HTTPS)	Platform licensing requests
Additional features (ports only required if the relevant feature is configured)					
SNMP server	<any>	Management Node / Conferencing Node	161	UDP	SNMP
Outlook client/add-in	<any>	Conferencing Node	443	TCP (HTTPS)	VMR Scheduling for Exchange
Management Node	55000–65535	FTP server	21 + server's FTP port range	TCP	FTP server for daily backup files
Management Node	55000–65535	LDAP server	389 / 636 3268 / 3269	TCP	LDAP AD global catalog searches
Management Node / Conferencing Node	55000–65535	Web proxy	8080 †	TCP	HTTP web proxy
Management Node / Conferencing Node	55000–65535	Incident reporting server (acr.pexip.com)	443	TCP (HTTPS)	Incident reporting ◊
Management Node	55000–65535	Usage statistics server (api.keen.io)	443	TCP (HTTPS)	Usage statistics ◊
Management Node	<any>	Exchange server	443	TCP (HTTPS)	VMR Scheduling for Exchange
Management Node	55000–65535	Cloud service	443	TCP (HTTPS)	Dynamic bursting to a cloud service provider ◊
Management Node	ephemeral	Teams Connector Azure Event Hub	5671/5672	AMQP	Only required if advanced status reporting is enabled in a Microsoft Teams integration
Management Node	55000–65535	SMTP server	587	TCP	SMTP (provisioning emails)

Source address	Source port	Destination address	Dest. port	Protocol	Notes
Management Node / Conferencing Node	<any>	SNMP NMS	161 †	UDP	SNMP Network Management System (NMS)
Management Node / Conferencing Node	55000–65535	Syslog server	514 †	UDP †	Syslog
Conferencing Node	55000–65535	Event sink server	80/443	TCP (HTTP/HTTPS)	Event sink
Conferencing Node	55000–65535	AD FS server	443	TCP (HTTPS)	Single Sign-On (SSO) with AD FS
Conferencing Node	55000–65535	Epic server	443	TCP (HTTPS)	Epic telehealth REST API requests ◊
Management Node	55000–65535	OAuth token endpoint <ul style="list-style-type: none"> for Exchange (when using OAuth for the service account): login.microsoftonline.com for Google Workspace domain user authorization: oauth2.googleapis.com/token for Webex-registered endpoints: webexapis.com 	<any> ‡	TCP (HTTPS)	One-Touch Join ◊
Conferencing Node	55000–65535	OAuth token endpoint <ul style="list-style-type: none"> for Exchange: login.microsoftonline.com for Google Workspace service account authorization: googleapis.com/oauth2/v4/token for Google Workspace domain user authorization: oauth2.googleapis.com/token for Webex-registered endpoints: webexapis.com 	<any> ‡	TCP (HTTPS)	One-Touch Join◊
Conferencing Node	55000–65535	Exchange Server	80/443 †	TCP (HTTP/HTTPS)	One-Touch Join ◊
Conferencing Node	55000–65535	Google Workspace	443 †	TCP (HTTPS)	One-Touch Join ◊
Conferencing Node	55000–65535	Cisco endpoint API	80/443 †	TCP (HTTP/HTTPS)	One-Touch Join ◊
Conferencing Node	55000–65535	Cisco Webex cloud (webexapis.com)	443 †	TCP (HTTPS)	One-Touch Join◊

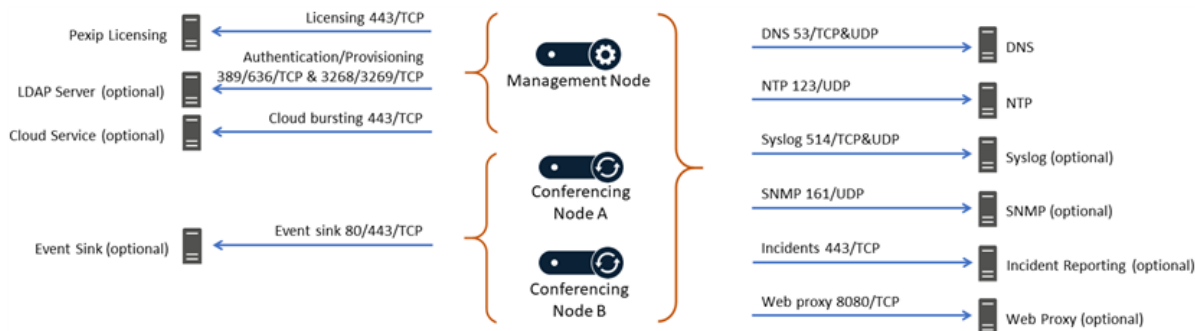
Source address	Source port	Destination address	Dest. port	Protocol	Notes
Poly endpoint	<any>	Conferencing Node	443	TCP (HTTPS)	One-Touch Join

† Configurable by the administrator.

‡ Determined by Exchange / Google Workspace.

◇ Does not apply if a [web proxy](#) has been configured.

Note also that the ephemeral port range (55000–65535) is subject to change.



Port requirements for peripheral services (only the most commonly-used services are shown)

Conferencing Node call signaling and media

These port usage rules for call signaling and media apply to Proxying Edge Nodes and Transcoding Conferencing Nodes:

Source address	Source port	Destination address	Dest. port	Protocol	Notes
Standard call signaling and media					
Endpoint	<any>	Conferencing Node	80	TCP (HTTP)	Web browser / API interface / Skype for Business / Lync system (for conference avatar)
Endpoint	<any>	Conferencing Node	443	TCP (HTTPS)	Web browser/ API interface / Infinity Connect mobile client
Endpoint / call control system	<any>	Conferencing Node	1719	UDP	H.323 (RAS signaling)
Endpoint / call control system	<any>	Conferencing Node	1720	TCP	H.323 (H.225/Q.931 signaling)
Endpoint / call control system	<any>	Conferencing Node	33000–39999 **	TCP	H.323 (H.245 signaling)
Endpoint / call control system	<any>	Conferencing Node	5060	TCP	SIP
Endpoint / call control system	<any>	Conferencing Node	5061	TCP	SIP/TLS

Source address	Source port	Destination address	Dest. port	Protocol	Notes
Endpoint / call control system	<any>	Conferencing Node	40000–49999 **	TCP/UDP	Endpoint / call control system / Skype for Business / Lync system / Infinity Connect++ RTP / RTCP / RDP / VbSS / DTLS / STUN / TURN
Conferencing Node	33000–39999 **	Endpoint / call control system	1719	UDP	H.323 (RAS signaling)
Conferencing Node	33000–39999 **	Endpoint / call control system	1720 / <any>	TCP	H.323 (H.225/Q.931 signaling) (to <any> if the device is registered to Pexip Infinity)
Conferencing Node	33000–39999 **	Endpoint / call control system	<any>	TCP	H.323 (H.245 signaling)
Conferencing Node	33000–39999 **	Endpoint / call control system	5060	TCP/UDP	SIP
Conferencing Node	33000–39999 **	Endpoint / call control system	5061	TCP	SIP/TLS
Conferencing Node	40000–49999 **	Endpoint / call control system	<any>	TCP/UDP	RTP / RTCP / RDP / VbSS / DTLS / STUN / TURN Endpoint / call control system / Skype for Business / Lync system / Infinity Connect++
Conferencing Node	40000–49999 **	STUN / TURN server	3478 †	UDP	STUN / TURN
Conferencing Node	40000–49999 **	RTMP streaming server	1935	TCP	RTMP streaming
Conferencing Node	55000–65535	SfB/Lync Web Conferencing service	443 / 8057 ‡‡	TCP (TLS)	PSOM (PowerPoint presentation from SfB/Lync)
Conferencing Node	55000–65535	SfB/Lync Front End Server or Edge Server and WAC/OWA/OOS server	443	TCP (TLS/HTTPS)	PowerPoint presentation from SfB/Lync
Additional features (ports only required if the relevant feature is configured)					
Endpoint / call control system	<any>	Conferencing Node	5060	UDP	SIP UDP
Google Meet	19302–19309	Conferencing Node	40000–49999 **	UDP	Google Meet SRTP/SRTCP
Client application	<any>	Conferencing Node	443	TCP (HTTPS)	Microsoft Teams (client application viewing the meeting invitation Alternative Dial Instructions)
Teams Connector	ephemeral	Conferencing Node	443	TCP	Microsoft Teams signaling
Teams Connector	50000-54999	Conferencing Node	40000–49999 **	UDP	Microsoft Teams SRTP/SRTCP

Source address	Source port	Destination address	Dest. port	Protocol	Notes
Conferencing Node	33000–39999 **	Google Meet (hangouts.clients6.google.com and meetings.googleapis.com)	443	TCP (HTTPS)	Google Meet
Conferencing Node	40000–49999 **	Google Meet	19302–19309	UDP	Google Meet SRTP/SRTCP
Conferencing Node	33000–39999 **	Teams Connector load balancer Teams Connector instance	443	TCP (HTTPS)	Microsoft Teams
Conferencing Node	40000–49999 **	Teams Connector instance	50000–54999	UDP	Microsoft Teams SRTP/SRTCP

† Configurable by the administrator.

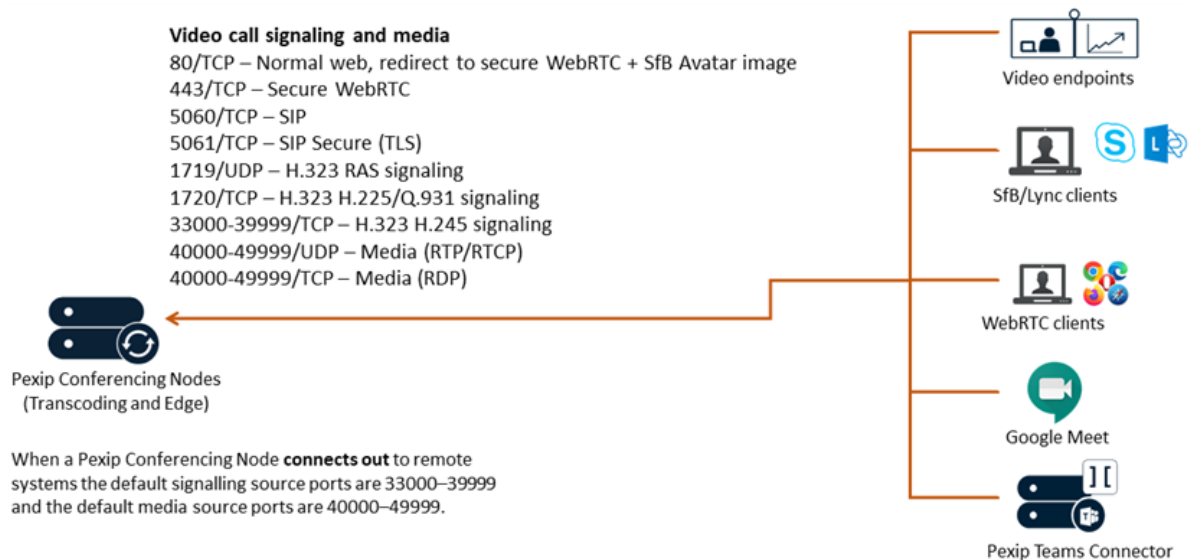
** Configurable via the Media port range start/end, and Signaling port range start/end options (see [About global settings](#)).

†† Infinity Connect web, mobile and desktop (installable) clients.

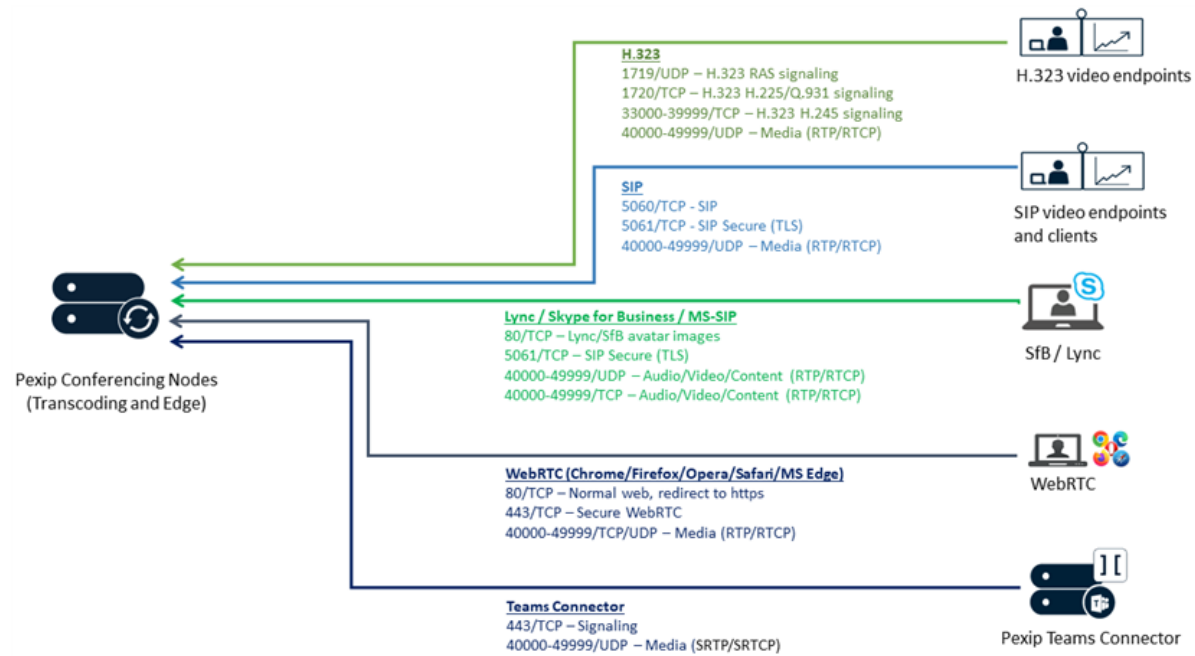
‡‡ Typically 443 for Web Conferencing Edge and 8057 for a Sfb/Lync Front End Server / FEP.

Note also that:

- ICE calls allocate 4 ports per media line/stream.
- The ephemeral port range (55000–65535) is subject to change.



Call signaling and media ports overview (inbound)



Call signaling and media ports details (inbound)

Troubleshooting the Pexip Infinity platform

This topic contains a list of symptoms, possible causes and suggested resolutions for some issues you may experience when using the Pexip Infinity platform.

It includes the following sections:

- [Pexip Infinity deployment and upgrading](#)
- [Dynamic bursting to a cloud service](#)
- [Joining a conference and viewing content](#)
- [Conference connectivity and TLS issues](#)
- [Pexip Infinity administration](#)
- [Infinity Connect clients](#)

For an up-to-date list of devices that are supported by Pexip Infinity, including any known issues, see [Interoperability](#).

Contacting support

If you cannot find the information you require, contact your Pexip authorized support representative. Technical support for software issues is available while under a valid support contract and running a Pexip Infinity version no more than 2 major software releases behind the current release. Software bug fixes will only be provided in either the current or the next major release of software.

Pexip Infinity deployment and upgrading

Symptom	Possible cause	Resolution
During upgrade, one or more Conferencing Nodes are stuck with a status of "waiting for calls to clear", but there are no active calls reported on the Management Node.	A completed call has not cleared properly from the Conferencing Node.	Reboot the Conferencing Node.
A Conferencing Node does not accept calls even though it is powered on and is contactable on the network.	If time is not properly synchronized between the Management Node and the host server, certificates issued by the Management Node may be invalidated by Conferencing Nodes within the same Pexip Infinity deployment. As a result, the Conferencing Nodes will not communicate properly with the Management Node, causing calls to fail.	<p>Ensure all virtual machines (i.e. the Management Node and all Conferencing Nodes) within the Pexip Infinity platform, and the host servers on which they are running, are using accurate times according to the public or private standard NTP clock.</p> <p>We strongly recommend that you configure at least 3 distinct NTP servers or NTP server pools in each instance to ensure proper synchronization.</p> <p>To synchronize time on Pexip Infinity:</p> <ol style="list-style-type: none">1. Synchronize time on the host servers (for instructions, see the relevant hypervisor installation guide).2. Enable NTP on Management Node.3. Reboot all VMs.

Symptom	Possible cause	Resolution
A newly deployed Conferencing Node does not accept calls and its last contacted status on the Management Node shows "Never", even though it is powered on and is contactable on the network.	After deploying a new Conferencing Node, it takes approximately 5 minutes before the node is available for conference hosting and for its status to be updated on the Management Node. Until it becomes available, the Management Node reports the status of the Conferencing Node as having a last contacted and last updated date of "Never". "Connectivity lost between nodes" alarms relating to that node may also appear temporarily.	Wait for the Conferencing Node to finish initializing.
The wrong information was entered while running the installation wizard.		<p>On server-based deployments you can re-run the installation wizard by following the instructions in Re-running the installation wizard.</p> <p>Do not re-run the installation wizard on cloud-based deployments (Azure, AWS, GCP or Oracle) in order to change Management Node configuration data such as its IP address or hostname. To change such data you must terminate the existing instance and deploy a new Management Node instance. You should only re-run the installation wizard on cloud-based deployments if you need to reset the web administration password (and then you should not change any of the other configuration data).</p>
A new Management Node or Conferencing Node does not work. It was created by cloning it through VMware.	You cannot use cloning to create Management Nodes or Conferencing Nodes.	<p>Create the Management Node according to our instructions (see Installation overview).</p> <p>Create all Conferencing Nodes by following the instructions in Deploying new Conferencing Nodes.</p>
A newly-deployed Conferencing Node has gone into maintenance mode with the message "CPU instruction set is not supported; system will be placed in maintenance mode".	The Conferencing Node has been installed on a system that does not meet the CPU instruction set requirements.	<p>Do one of the following:</p> <ul style="list-style-type: none"> • Migrate the Conferencing Node to another server. • Re-install the Conferencing Node on another server. • If the instruction set is limited because you are using Enhanced vMotion Compatibility (EVC) (or equivalent), increase the minimum EVC level to L4 (Sandy Bridge).

Dynamic bursting to a cloud service

Symptom	Possible cause	Resolution
No bursting nodes appear in the Cloud overflow nodes area of the Status > Cloud Bursting page	The cloud node instances are not tagged correctly.	Check that the node instances running in your service provider have been assigned the pexip-cloud tag and that the tag value is set to the Management Node hostname.

Symptom	Possible cause	Resolution
You see a status issue "Instance <name> (with IP <address>) was found, but no corresponding Conferencing Node has been configured".	This occurs when Pexip Infinity detects a bursting instance with a tag matching your system's hostname but there is no corresponding Conferencing Node configured within Pexip Infinity.	This message can occur temporarily in a normal scenario when deploying a new Conferencing Node and you have set up the VM instance in your cloud provider but you have not yet deployed the Conferencing Node in Pexip Infinity. In this case, the issue will disappear as soon as the Conferencing Node is deployed.
You see connectivity errors in the administrator log while overflow nodes are being started/stopped.	This is normal behavior.	No action required.
"Not authorized to perform an operation on <instance ID or region name>. Check the policy created for the AWS user." error.	This means that there is a problem with the AWS policy document, or the AWS user is not attached to the policy.	See Deploying Pexip Infinity on Amazon Web Services for more information.
You see a status issue "Cloud bursting process encountered the following error: unsupported operand type(s) for -: 'datetime.datetime' and 'NoneType'"	A system location's configured Transcoding location only contains bursting nodes.	Ensure that the system location's configured Transcoding location contains "always-on" nodes.

Joining a conference and viewing content

Symptom	Possible cause	Resolution
<p>Participants cannot join a conference due to insufficient capacity.</p> <ul style="list-style-type: none"> When users attempt to join a conference they get a message saying "Participants cannot join a conference due to insufficient capacity." There is an alarm "Call capacity limit reached". The administrator log is reporting "Participant failed to join conference" and "out of proxying resource" or "out of transcoding resource". 	<p>A call has not been accepted because all Conferencing Nodes that are able to take the media for this call are at capacity. It could be either Proxying Edge Nodes or Transcoding Conferencing Nodes that are out of capacity.</p>	<ul style="list-style-type: none"> Deploy more Conferencing Nodes in either the proxying or transcoding location as appropriate. Move existing Conferencing Nodes onto more powerful servers. Allocate more virtual CPUs for Conferencing Nodes on existing servers (if there are sufficient CPU cores). Note that the Conferencing Node will have to be rebooted for this to take effect. Configure each location with a primary and secondary overflow location. If a call is received in a location that contains Proxying Edge Nodes, that location must be configured with a Transcoding location that contains your Transcoding Conferencing Nodes. <p>Note that some types of call consume more resources than other calls. Thus, for example, if you are at full capacity and an audio-only call disconnects, there may still not be sufficient free resource to connect a new HD video call.</p>
	<p>The participant has dialed in to the Conferencing Node while it is still starting up and an internal capacity-checking tool is running.</p>	<p>Wait for one minute and then attempt to join the conference.</p>
	<p>The Virtual Meeting Room or Virtual Auditorium has a participant limit applied, and this limit has been reached.</p>	<p>Increase the participant limit, if appropriate.</p>
<p>Participants cannot join a conference due to an invalid license.</p> <ul style="list-style-type: none"> When users attempt to join a conference they get a message saying "Participants cannot join a conference due to an invalid license." The admin log is reporting "Participant failed to join conference" and "no valid license available". 	<p>If your Pexip Infinity reports an invalid license, this could mean that:</p> <ul style="list-style-type: none"> the license has not been activated the existing license has expired the existing license has become corrupt (this could occur, for example, if the Management Node reboots after an upgrade and comes back up on a different physical blade with a new MAC address). 	<ul style="list-style-type: none"> Check the status of your licenses from the Licensing page (Platform > Licenses). Contact your Pexip authorized support representative for assistance. <p>For more information, see Pexip Infinity license installation and usage.</p>

Symptom	Possible cause	Resolution
<p>Participants cannot join a conference due to insufficient licenses.</p> <ul style="list-style-type: none"> When users attempt to join a conference they get a message saying "Participants cannot join a conference due to insufficient licenses." There is an alarm saying "License limit reached". The admin log is reporting "Participant failed to join conference" and "license limit reached". 	<p>There are not enough call licenses available on the system at this time. For more information, see Pexip Infinity license installation and usage.</p>	<ul style="list-style-type: none"> Wait until one or more of the existing conferences have finished and the call licenses have been returned to the pool. Contact your Pexip authorized support representative to purchase more call licenses.
<p>An H.323 endpoint has its bandwidth restricted when joining a conference via a Virtual Reception, or when placing a call using the Distributed Gateway after first connecting to a Virtual Reception.</p>	<p>If there has been a bandwidth restriction placed on the Virtual Reception, any H.323 endpoints using that service will not be able to subsequently increase their bandwidth, even after being transferred to a Virtual Meeting Room or using a Call Routing Rule that has a higher (or no) limit.</p>	<ul style="list-style-type: none"> Make the call using a SIP endpoint or Infinity Connect client. Do not place a bandwidth restriction on the Virtual Reception.
<p>Presentations do not display full screen.</p>	<p>If the presentation being shared is either:</p> <ul style="list-style-type: none"> an application that is not in full-screen mode a full screen image that is being sent from a non-standard aspect ratio screen <p>then the image being sent may have a non-standard aspect ratio. To send the image inside a standard resolution window (for example 640x480 [4:3]) or 1280x720 [16:9]), the endpoint may add horizontal or vertical mattes (known as letterboxing or pillarboxing respectively).</p>	<p>Ensure that presenters always either:</p> <ul style="list-style-type: none"> share their entire screen, or share individual applications when they are in full-screen mode only.
<p>Images are not displaying as expected:</p> <ul style="list-style-type: none"> they are being cropped they have black bars at the top or sides 	<p>Endpoints send and display video images and presentations in various aspect ratios, most commonly 16:9 and 4:3. If there is a difference between the aspect ratios of the sending and receiving endpoints, then the endpoint and/or Pexip Infinity may crop the image or add vertical or horizontal mattes.</p>	<p>For more information, see Changing aspect ratios.</p>
<p>Occasional video freezes are seen in media received from SIP or H323 endpoints, where Conferencing Nodes in GCP (including PSS nodes) are in use.</p>	<p>System locations that include Conferencing Nodes running in Google Cloud Platform (including Pexip Smart Scale locations) must be configured with a maximum MTU of 1460 bytes. Some older endpoints default to a MTU higher than this.</p>	<p>Reduce the MTU on the endpoint to 1350.</p>

Symptom	Possible cause	Resolution
<p>Main video on a Cisco E20 freezes and no presentation is shown when a VMR participant starts presenting.</p> <p>The E20 still sends video as normal, and audio flows in both directions.</p>	<p>This may occur when additional video codecs (e.g. H.264 High Profile) are enabled on the Pexip Infinity deployment, and if the E20 is called from a VMR and then a participant in the VMR presents content.</p> <p>This occurs because more codecs are offered to the E20 than it can cope with.</p>	<p>Disable all video codecs (i.e. H.264 High Profile) not required in the deployment (Platform > Global Settings > Codecs).</p>
<p>Cisco endpoints running TC version 5.x or earlier software do not receive video when dialed out to from Pexip Infinity. Video is still sent as normal.</p>	<p>This may occur when additional video codecs (e.g. H.264 High Profile) are enabled on the Pexip Infinity deployment, and the TC 5.x or older is called from a VMR.</p> <p>This occurs because more codecs are offered to the TC endpoint than it can cope with.</p>	<ul style="list-style-type: none"> Upgrade the TC 5.x endpoint to a more recent software version. Disable all video codecs (i.e. H.264 High Profile) not required in the deployment (Platform > Global Settings > Codecs).
<p>A Cisco endpoint running TC 7.1.4 might not receive video from Pexip in H.323 calls.</p>	<p>This is due to a bug in TC 7.1.4 software.</p>	<ul style="list-style-type: none"> Disconnect and reconnect the call. Use SIP instead of H.323. Upgrade the TC software.
<p>Cisco endpoints running TC 7.0.x or earlier may crash when connecting to or from Pexip Infinity v24 or later over SIP.</p>	<p>This happens when AES-256 cipher is included in the SDP.</p>	<p>Upgrade the TC software to the latest version.</p>
<p>Issues with Polycom endpoints receiving any presentation >1080 lines tall, such as 1600x1200 (UXGA) presentations.</p>	<ul style="list-style-type: none"> A Polycom HDX series may crash and require multiple reboots after being sent a presentation >1080 lines tall. A Polycom RealPresence Group series endpoint may stop receiving presentation for the rest of the call after being sent a presentation >1080 lines tall. 	<p>Put the endpoint in 720p mode.</p>
<p>Participants keep hearing themselves repeated back after a short delay. This happens in a conference with one or more other participants connected using Infinity Connect via Edge, Safari or Firefox, and who are using their computer's microphone and speakers.</p>	<p>Edge, Safari and Firefox do not have adequate echo cancellation, and in certain circumstances may experience a delay in playing audio. When this happens, sounds played through the computer's speakers are picked up by the computer's microphone and replayed back to other participants.</p>	<p>Participants using Edge, Safari or Firefox should:</p> <ul style="list-style-type: none"> use a headset mute themselves when not speaking consider using Chrome or the Infinity Connect desktop client instead.
<p>In-band DTMF tones may not be detected if they are input too quickly.</p>	<p>False detections can be caused, for example, by poor line quality, line noise and echo.</p>	<p>This is best resolved through using out-of-band DTMF tones.</p>
<p>Participants are disconnected from conferences and "Backplane disconnected" messages are recorded in the administrator log.</p>	<p>VMware snapshots were being taken or deleted while conferences were in progress. Taking or removing snapshots can cause virtual machines to become unresponsive.</p> <p>For more information, see this VMware knowledge base article.</p>	<p>Only create and delete VMware snapshots at a time of minimal usage.</p>

Conference connectivity and TLS issues

Symptom	Possible cause	Resolution
Calls fail when dialing out to, or receiving calls from, video network infrastructure devices, such as a Cisco VCS or CUCM. Calls may fail immediately, or after a period of time, and SSL alerts are raised in the support log with an "unsupported certificate" description.	Either the external system or the Conferencing Node is verifying the other party's certificate and it is rejecting the connection because the certificate does not have client authentication properties.	Ensure that the certificates on your external systems and on your Conferencing Nodes contain "TLS Web Client Authentication" Enhanced Key Usage properties (see Mutual TLS authentication and client/server certificates).
A Cisco MXP intermittently puts a call on hold immediately after resuming it.	This is due to a bug in the MXP where a race condition exists between the resume message and the session refresh re-INVITE message.	Set the session refresh configuration on your call control system to a value that avoids this race occurring.

TLS certificate administration

Error message	Possible cause	Resolution
Certificate and private key do not appear to be part of the same key pair	This most likely means that you have tried to upload the certificate against the wrong CSR.	Select the correct CSR and try again.

Pexip Infinity administration

Symptom	Possible cause	Resolution
A policy profile is configured but it is being ignored.	The policy profile has not been assigned to a system location.	Assign the policy profile to your locations (Platform > Locations).
Log timestamps appear to be inaccurate or log entries appear to be out of sequence.	Time is not properly synchronized between the Management Node, Conferencing Nodes and their host servers, causing different systems to use different timestamps. This could be because: <ul style="list-style-type: none"> Insufficient NTP servers or NTP pools have been configured on a host server or the Management Node (we recommend a minimum of 3). One or more NTP servers are unreachable or have inaccurate time themselves. NTP is otherwise not configured according to our recommendations. 	<p>Ensure all virtual machines (i.e. the Management Node and all Conferencing Nodes) within the Pexip Infinity platform, and the host servers on which they are running, are using accurate times according to the public or private standard NTP clock.</p> <p>We strongly recommend that you configure at least 3 distinct NTP servers or NTP server pools in each instance to ensure proper synchronization.</p> <p>To synchronize time on Pexip Infinity:</p> <ol style="list-style-type: none"> 1. Synchronize time on the host servers (for instructions, see the relevant hypervisor installation guide). 2. Enable NTP on Management Node. 3. Reboot all VMs.
How do I register Pexip Infinity to a gatekeeper?		You don't register Pexip Infinity to a gatekeeper. Instead, configure your call control system to route calls to Pexip Infinity. See Call control .

Symptom	Possible cause	Resolution
Oracle Acme Packet SBC has the error "Message Too Large".		Increase the following configuration parameters on the Acme Packet SBC: <ul style="list-style-type: none"> <code>sip-message-len 16000</code> <code>option +max-udp-length=0</code>
VMware datastore is showing disk I/O alarms.		Enabling SIOC on your datastores might help. For more information, see this VMware knowledge base article .
A participant's Display name has some characters replaced with asterisks.	For H.323 endpoints, the display name should use IA5 characters, but some endpoints use UTF-8 or other encodings. If Pexip Infinity receives any non-IA5 characters, it replaces them with an asterisk.	Ensure the display name used by such endpoints uses a string that contains valid IA5 characters only.
Unable to take snapshots on deployments hosted in Microsoft Azure.	The Azure Load Balancer is timing out.	Increase the TCP idle timeout setting in Azure for the Azure Load Balancer. See https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-tcp-idle-timeout for configuration instructions.

Infinity Connect clients

The table below describes general issues that may occur when using Infinity Connect clients within your deployment. For a list of specific error messages that may be presented to Infinity Connect users, along with their meaning and suggested resolution (where appropriate), see [Troubleshooting Infinity Connect error messages](#).

Symptom	Possible cause	Resolution
Participants cannot use Infinity Connect clients. Web app and desktop client users are presented with a message "This feature has been disabled."	Support for these Infinity Connect applications has been disabled.	Enable support as follows: <ol style="list-style-type: none"> Go to Platform > Global Settings > Connectivity. Select the Enable support for Pexip Infinity Connect clients and Client API checkbox.
Participants attempting to connect to a conference using the Infinity Connect mobile client for iOS get a message "Certificate Error. Please contact your system administrator."	Your deployment does not use valid, trusted certificates. Infinity Connect mobile clients require deployments with HTTPS and valid certificates.	Install valid certificates. For more information, see Managing TLS and trusted CA certificates .
Participants using an Infinity Connect client cannot write or see chat messages.	Chat has been globally disabled.	Enable chat .

Symptom	Possible cause	Resolution
Guest participants using an Infinity Connect client cannot see the chat window, participant list, or presentation.	Guest participants are not allowed in to the conference if the conference is locked, or until the first Host participant has joined.	<ul style="list-style-type: none"> • Allow an individual Guest to join the locked conference. • Unlock the conference, allowing all Guests to join. • Connect at least one endpoint to the conference as a Host participant either as: <ul style="list-style-type: none"> ◦ a video or audio participant, or ◦ an Infinity Connect presentation and control-only participant, and then start the conference manually.
Users of the Infinity Connect web app see "An error occurred. The page you are looking for is currently unavailable." error message when connecting via a reverse proxy.	You are running version 3 or earlier of the Pexip Reverse Proxy and TURN Server and the web browser cannot make a TLS connection.	Upgrade to latest version of the Pexip Reverse Proxy and TURN Server.
Users see a Connection Alert message at the top of their window	There is packet loss of 3% or more between the Infinity Connect client and Pexip Infinity.	Users who experience this issue regularly should select a lower default bandwidth.

