Teradici PCoIP Graphics Agent for macOS 22.04

Always have a secondary means of access to your remote machine

Currently, the Graphics Agent for macOS can be impacted by operating system-level events which may cause PCoIP connections to stop working. To recover from these situations, you must be able to access the remote machine directly, either physically or via VNC. See Troubleshooting for help.

This guide is intended for administrators who are deploying the Graphics Agent for macOS as part of the Teradici Cloud Access Software. It assumes thorough knowledge of macOS conventions and general networking concepts, including firewall configuration.

About the Graphics Agent for macOS 22.04

The Graphics Agent for macOS is part of Teradici Cloud Access Software. It enables Teradici customers to deliver GPU-powered physical workstations to end users via remote clients.

Users include high-end knowledge workers, graphic designers, artists, and CAD/CAM designers.

A deployed Graphics Agent for macOS requires these components:

- **A host machine** which provides the desktop to remote clients. The host must be supported macOS machine. See System Requirements for more information.
- **The agent software** installed on the host machine.

About GPUs

The Graphics Agent for macOS uses the GPU in your Mac. You do not need to install a secondary GPU.

Where to Find Information about Other Components

This guide describes the Graphics Agent for macOS.

For complete information about all of the components used in PCoIP ecosystems, including architectural diagrams and deployment suggestions, see one of the following documents:

Cloud Access Software architectures and descriptions:

- Teradici All Access Architecture Guide

For more information about PCoIP clients, see one of the following:

- Teradici PCoIP Software Client 22.04 for Windows Administrators' Guide
- Teradici PCoIP Software Client 22.04 for macOS Administrators' Guide
- Teradici PCoIP Software Client for 22.04 Linux Administrators' Guide
- Tera2 PCoIP Zero Client 22.04 Administrators' Guide

For information about Cloud Access licensing, see our Licensing FAQ. Most PCoIP systems use PCoIP Cloud Licensing. For systems using a local PCoIP License server instead, refer to the following guides:

- Teradici PCoIP License Server Administrators' Guide for ***Online Environments***
- Teradici PCoIP License Server Administrators' Guide for ***Offline Environments***

What's New in This Release

Release 22.04 of the Graphics Agent for macOS includes:

- **PCoIP Ultra Collaboration** now supports CAS-brokered connections, providing the ability to share your PCoIP session with a remote collaborator using the PCoIP Connection Manager 22.01+. See Feature Support: Collaboration for details.
- Adds support for macOS 12 (Monterey). Removes support for macOS 10.15 (Catalina).
- Bug fixes and stability enhancements.

System Requirements

Important: M1 processors are not currently supported

The Graphics Agent for macOS requires an Intel processor. M1 processors are not currently supported.

The Graphics Agent for macOS depends on the following system capacities and capabilities:

Host Requirements

Host machine requirements	
Operating Systems	• macOS 11 (Big Sur) or macOS12 (Monterey)
Hardware Configurations	Supported on newer (2017+) models of Intel-based Mac hardware, including: • Mac Pro • Mac Mini • iMac Pro • iMac • MacBook Pro Models older than 2017 may work, but have not been tested and are not supported.
Remote Host Memory	• 8GB or greater

Host machine requirements	
Network Ports	The following ports must be open on the host desktop: • TCP 443 • TCP 4172 • UDP 4172 • TCP 60443 Collaboration sessions require an open UDP port (default 64172) For **PCoIP Ultra Collaboration**, incoming traffic on **UDP 4173** must also be open.
Storage	At least 100MB for installation and 100MB for logging are recommended.
User	You must create a user account for PCoIP connections. The user name cannot contain spaces, and cannot be the ***root user*** account.

Important: Many factors can impact performance

Actual performance depends on several factors, including available bandwidth, client and host machine capabilities, the number of monitors in use, and workload requirements.

Audio Support

The Graphics Agent for macOS supports audio input and output between the host and the client. To send audio to the PCoIP client, you must select the **PCoIP Virtual Speaker** audio device as the output device on the remote macOS system.

Audio can be enabled, disabled, and audio bandwidth throttled by configuring the agent.

Multi-Channel Audio Output

Requirements

- **Client**: PCoIP Client for macOS, version 22.01 or newer

> **Important: macOS Client is required**
>
> Multi-channel audio is only supported by the PCoIP Client for macOS.

- **Agent**: Graphics Agent for macOS, version 22.01 or newer
- **Audio device**: Multi-Channel Audio device that supports 2.1, 5.1 or 7.1 channel configuration (connected to PCoIP Client for macOS)

Current Limitations

- Only 2.1, 5.1, and 7.1 configurations are currently supported.

Enabling Multi-Channel Audio

To use multi-channel audio in a PCoIP session, the client Mac must be connected to a multi-channel audio device that is set as the system default. When a configured client establishes a PCoIP connection, the Graphics Agent for macOS will automatically detect the multi-channel system.

See the PCoIP Software Client for macOS Administrators' Guide for setup instructions.

Collaboration

The PCoIP Ultra Collaboration feature enables a PCoIP session user to share their session with a remote guest collaborator using standard PCoIP Soft Clients. While connected the guest collaborator can view the screen output and hear the audio output of the shared PCoIP session.

When discussing this feature, we'll refer to the first user as the ***host collaborator***, and the second user who joins the session as the ***guest collaborator***.

Requirements

- The Collaboration feature must be hosted on a Graphics Agent for macOS 22.04 or higher, with ***Collaboration*** and ***PCoIP Ultra CPU Offload*** enabled.
- Both the host and the guest collaborators must connect using a PCoIP Software Client 22.04 or higher (macOS, Windows, or Linux).
- Both collaborators must connect using PCoIP software clients that support ***PCoIP Ultra CPU Offload***.
- Collaboration sessions use a UDP port which must allow inbound traffic, both at the cloud provider network level and the local firewall. **The default collaboration port is UDP 64172**; if necessary, this can be changed. See Changing the collaboration port for details.
- For ***brokered collaboration***, the PCoIP Connection Manager and PCoIP Security Gateway 22.04 or later is required, and:
    - If the brokered connection is via the PCoIP Security Gateway, then ***the PCoIP Security Gateway*** must be able to connect to the host on the configured collaboration port (UDP 64172 by default).
    - If the brokered connection is ***not*** via a PCoIP Security Gateway, then ***the guest collaborator's PCoIP client*** must be able to connect to the host on the configured collaboration port (UDP 64172 by default).
- For ***unbrokered (direct) collaboration***, the guest collaborator's PCoIP client must be able to connect to the host on the configured collaboration port (UDP 64172 by default).

Current Limitations

- Only one guest collaborator can connect at a time.
- Collaboration sessions support only one screen. The host collaborator should set their PCoIP Software Client to ***Fullscreen One Monitor*** mode prior to starting the collaboration session.
- If the host and guest screen resolutions are different, the guest's screen will use scrollbars and letterboxing to display the shared content.

  If ***high performance client*** mode is enabled, and if the host's resolution is greater than the guest's, the guest's screen will be clipped instead.
- The guest collaborator's session can only view and listen to the shared session. The guest collaborator has no ability to control the host's keyboard, mouse, microphone, or any other input device.
- The guest collaborator will not see the mouse cursor in the shared session.
- Collaboration session tokens expire after 1 hour. The expiration time is not currently configurable.
- Collaboration session tokens are single use. Once a collaboration guest has connected, a new token must be generated.
- When a collaboration session is disconnected by the guest collaborator, the ***Stop Collaboration*** button in the Collaboration Management console may incorrectly remain enabled. If this occurs, click **Stop Collaboration** to reset the button state and allow a new collaboration session to be started.
- Collaboration using PCoIP Ultra GPU Offload and Auto Offload are supported as experimental features on the PCoIP Graphics Agent only. PCoIP Ultra GPU Offload and Auto Offload are not supported on the PCoIP Standard Agent.

Enabling Collaboration

The PCoIP Ultra Collaboration feature is disabled by default. To enable this feature, both ***PCoIP Ultra CPU Offload*** and ***Collaboration*** must be activated on the Graphics Agent for macOS.

1. Open a Terminal window and enter the following commands:

```
sudo defaults write /Library/Preferences/com.teradici.pcoip-agent.plist
pcoip.enable_collaboration 1
sudo defaults write /Library/Preferences/com.teradici.pcoip-agent.plist
pcoip.ultra 1
```

2. The default collaboration port is UDP 64172. If you need to change it, provide the new port number with an additional Terminal command:

```
sudo defaults write /Library/Preferences/com.teradici.pcoip-agent.plist
pcoip.collaboration_udpport <new_collaborator_port>
```

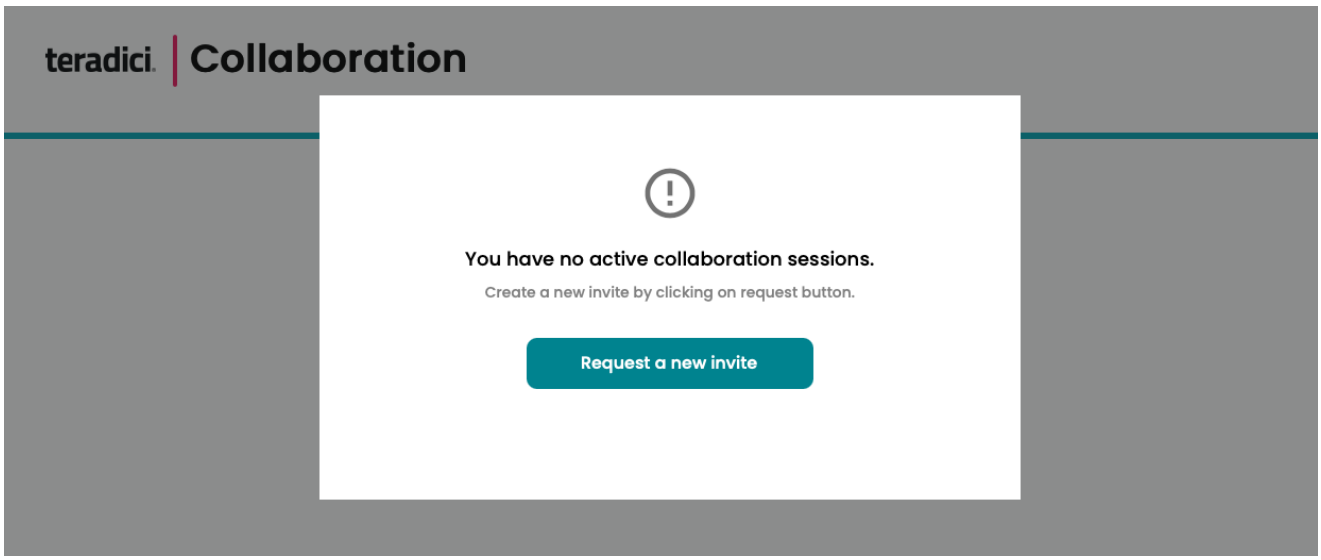
3. Reboot the Mac.

See Configuration Guide - Configurable Settings for more detailed information on setting configuration values.

Hosting a Collaboration Session

To host a PCoIP Ultra Collaboration session, the host collaborator starts a PCoIP session, then generates an invitation token that is passed to the guest collaborator:

1. Connect to a PCoIP session with PCoIP Ultra CPU Offload enabled.
2. Open the *Collaboration Management Console* by pressing ⌘+`spacebar` and typing **PCoIP Collaboration**, or by locating **PCoIP Collaboration** app in the *Applications* folder.
3. In the Collaboration Management Console, click **Request a new invite**.

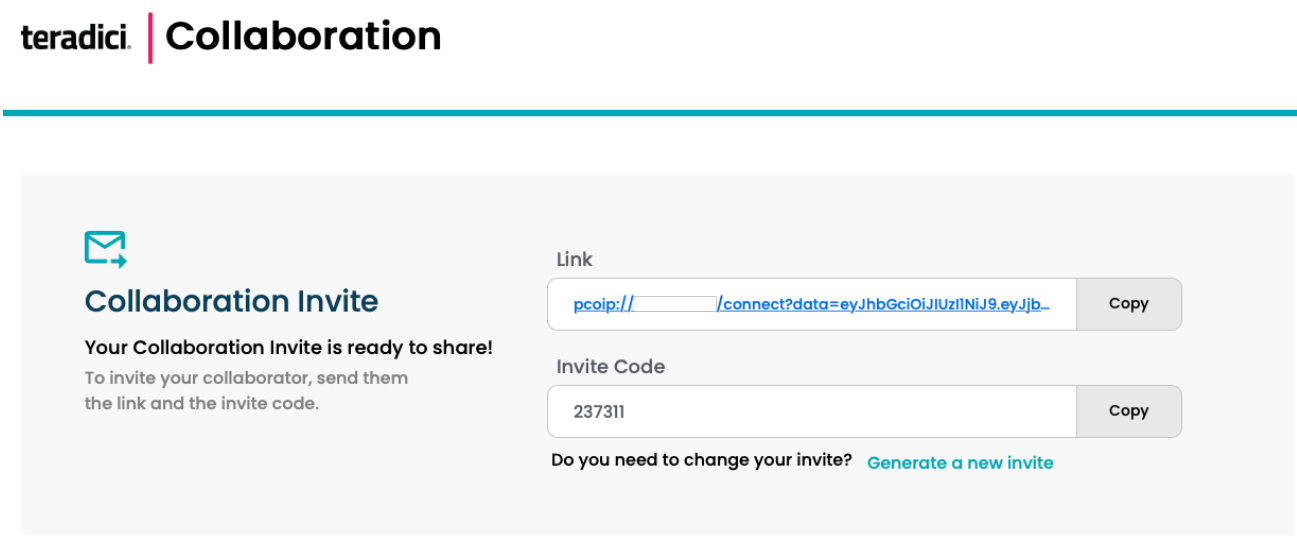


Generating a new link and invite code

If you have already generated an invite but need to create a new one, click **Stop Collaboration** to invalidate the first invite and then click **Start Collaboration** to create a new one.

4. The Collaboration Management Console provides two pieces of information that are used to invite the guest collaborator:
   - **Link**: The guest collaborator will use this link to join your session. This URI may be opened on any Mac, Windows or Linux machine with a PCoIP Software Client 21.03 or newer.

     This URI contains a collaboration token which will expire *1 hour after the Host session was established*. The generated URI can only be used once. If the token expires, a new invite must be generated.
   - **Invite Code**: This is a 6-digit code that confirms the identity of the individual connecting to the collaboration session. A new code is generated along with each new token.

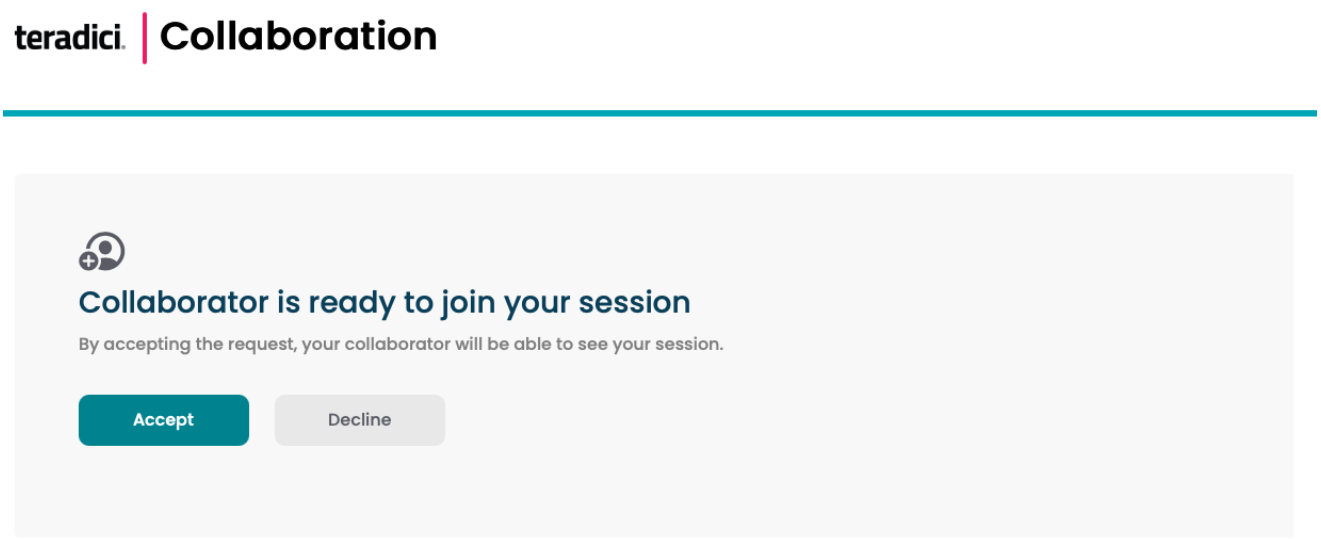


5. Share the PCoIP URI and the Collaboration Invitation Code with the guest collaborator.

Security best practice

Teradici recommends that these two pieces of information be shared with the guest user in separate communications, reducing risk in the event that a message is inadvertently sent, forwarded, or intercepted by a third party.

6. When the guest collaborator attempts to join the session, the Collaboration Management Console will display options to accept or reject the connection.



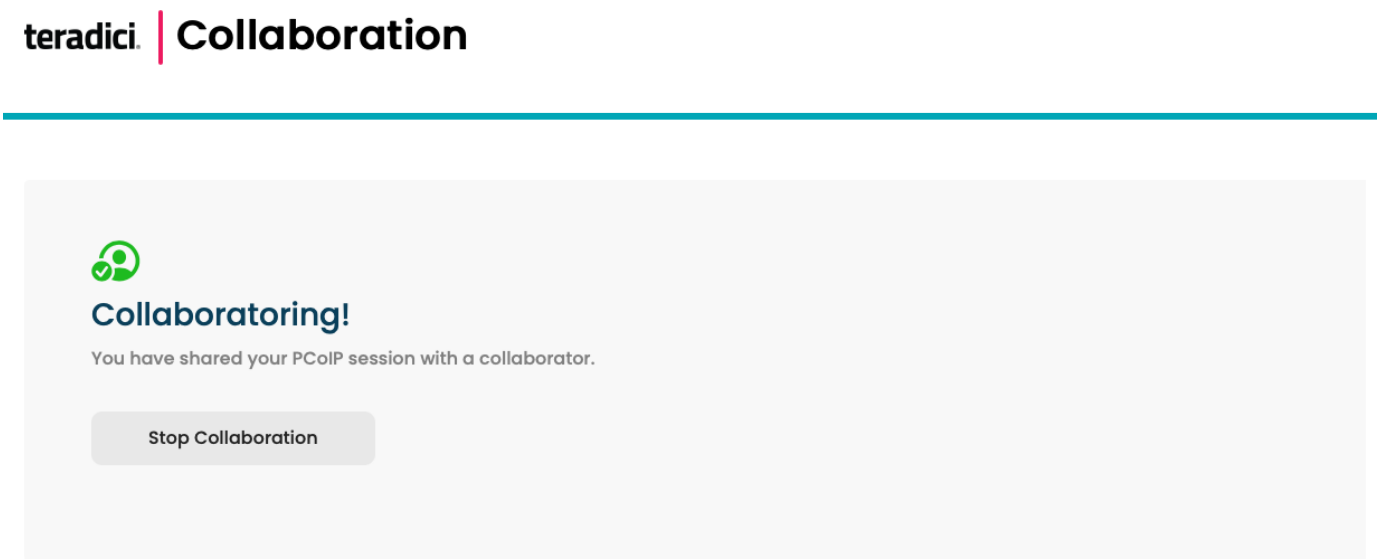
7. Click **Accept** to start the collaboration session. Click **Decline** to deny the request. Whether you accept the request or not, the invite has been used and is now disabled. Subsequent attempts will require a new invite.

Ending a Collaboration Session

The collaboration session will end when the host stops collaborating, or if either the host or guest collaborator disconnects their PCoIP connection.

To stop collaborating with a guest:

In the Collaboration Management Console, click the **Stop Collaboration** button. This ends the collaboration session. Once the session ends, the host can request a new invite and repeat the process above to start a new session.



Joining a Collaboration Session

The guest collaborator can join the session once they have received the PCoIP URI and the Collaboration Invitation code.

1. Open a web browser and go to the PCoIP URI shared with you (you may be able to click this link directly, depending on how it was shared with you).
2. The web browser will warn you that the link is attempting to open the ***PCoIP Client*** application. Allow the browser to open the PCoIP client.
3. When the PCoIP client opens, it will prompt you for your name and the Collaboration Invitation Code. The value you enter for your name is used to tell the host who is joining; the Collaboration Invitation Code is the six digit number provided by the host. Enter both values and click **Submit**.
4. Once the host collaborator accepts your connection request, the Collaboaration screen share will start.
5. To leave the collaboration session, select **Connection** > **Disconnect** from the PCoIP Client menu.

Changing the Collaboration Session Port

The default UDP Port for collaborator sessions is 64172. If necessary, you can change this port.

To change the Collaboration session port:

1. Open a Terminal window, and type the following command specifying the new UDP port number to use:

```
sudo defaults write /Library/Preferences/com.teradici.pcoip-agent.plist
pcoip.collaboration_udpport <new_collaborator_port>
```

2. Reboot the machine.

Supported Displays

The Graphics Agent for macOS supports a maximum of four displays on the PCoIP client and a maximum resolution of 4K UHD (3840×2160).

If you are using a monitor that has more than 3840 pixels in either dimension, you must scale your display to 3840x2160 or lower ***before*** launching the PCoIP Client.

Connected monitors, resolution scaling, and rotation should all be configured and fixed on the client computer ***before*** establishing a remote PCoIP session. Changing screen display scaling, rotation or number of displays during a PCoIP session is not supported, and can result in unexpected display or mouse behavior.

> **Important: Local displays are disabled when in session**
>
> When a PCoIP session begins, all physical displays attached to the host machine—including built-in and connected external monitors—will show black screens. After the session ends, the host physical displays are re-enabled, and function as usual.

PCoIP agents support any of these monitor configurations:

- Vertical line
- Horizontal line
- Box display

Monitors can be used in any standard rotation (0°, 90°, 180°, or 270°). Any monitor can be the primary.

The Graphics Agent for macOS provides the following benefits:

- **Display resolutions**: The Graphics Agent for macOS can provide any resolution a client asks for up to 4K UHD.
- **3D application support**: Full-screen 3D applications are supported.

Note: Using multiple high-resolution displays

Systems with multiple high-resolution displays, such as quad 4K UHD topologies, require powerful system infrastructure. Be sure to use a system with sufficient bandwidth, client capabilities, and host capabilities to support your required display topology.

PCoIP Ultra

The Graphics Agent for macOS provides support for PCoIP Ultra, the latest protocol enhancements from Teradici. PCoIP Ultra is optimized for truly lossless support with bit-exact color accuracy and preservation of content detail at the highest frame rates.

PCoIP Ultra protocol enhancements propels our industry-recognized performance into the future of remote computing, with faster, more interactive experience for users of remote workstations working with high-resolution content.

PCoIP Ultra enhancements are disabled by default. You must enable them manually.

PCoIP Ultra is appropriate for specific use cases

For most users, the default PCoIP protocol will provide the best possible experience. Carefully review the recommended use cases in the next section to determine whether you should enable it.

For additional detail on PCoIP Ultra technical requirements for various use cases and troubleshooting steps, refer to KB 2109: PCoIP Ultra Troubleshooting.

When to Enable PCoIP Ultra

PCoIP Ultra is apporpriate for users with the following requirements:

Auto Offload: Achieves the best balance between color accuracy and network efficiency. This setting is appropriate for work-from-home or WAN content creators who require optimized delivery of high resolution content, including video playback, while still achieving build-to-lossless color accuracy.

CPU Offload: Provides efficient scaling across multicore CPUs, leveraging AVX2 instruction sets. Appropriate for users that require CPU-optimized delivery of 4K UHD, high-framerate video playback and build-to-lossless color accuracy. It is also useful when GPU encoding resources must be reserved for video encoding applications, typically in LAN environments.

GPU Offload: PCoIP encoding is always offloaded to a GPU. Appropriate for users who demand the highest possible CPU efficiency.

For ***all other scenarios***, Teradici recommends that you leave PCoIP Ultra disabled.

Requirements

To take advantage of PCoIP Ultra, you need:

- A **PCoIP agent** (any type), 22.04 or later
- A **PCoIP Software Client** (any type), 22.04 or later

> ⚠ **PCoIP Tera2 Zero Clients do not support PCoIP Ultra**
>
> PCoIP Ultra is only available through PCoIP Software Clients.

- **CPU offload** requires CPU support for the AVX2 instruction set on both the agent and client machines.
- **GPU offload** requires an NVIDIA graphics card that supports NVENC on the agent machine.

Enabling PCoIP Ultra

To enable PCoIP Ultra features, open a Terminal window and enter the following command:

```
sudo defaults write "/Library/Preferences/com.teradici.pcoip-agent.plist"
pcoip.ultra <ultra_mode_setting>;
```

...where `<ultra_mode_setting>` is one of the following:

- 1: *PCoIP Ultra CPU Offload*. CPU offload requires CPU support for the AVX2 instruction set on both the remote host and client. The PCoIP Zero client is not supported. CPU offload is recommended for 4K UHD resolutions with video playback requirements of 30 fps (or more), and the highest possible image quality and color accuracy.
- 2: *PCoIP Ultra GPU Offload*. GPU offload requires an NVIDIA graphics card on the remote host capable of NVENC. GPU Offload is recommended when the CPU impact of pixel encoding should be minimized.

- 3: *PCoIP Ultra Auto Offload*. This setting allows PCoIP to automatically switch between CPU and GPU Offload modes; CPU offload is used by default to provide the best image fidelity, and GPU offload is used during periods of high display activity to provide improved frame rates and bandwidth optimization.

  This setting is only effective if the remote host and client endpoints are capable of both CPU and GPU Offload.

All PCoIP Ultra settings take effect on the next PCoIP session. No configuration is required on the PCoIP Software Client.

Turning PCoIP Ultra off

To disable PCoIP Ultra and use the default PCoIP experience instead, set `pcoip.ultra` to 0.

Setting configuration values

If you don't know how to set PCoIP agent configuration values, refer to Configuring the Graphics Agent for macOS.

Printing Support

The Graphics Agent for macOS does not support local printing on remote clients.

Local, network, and cloud printers are supported in various ways:

- macOS hosts can print to any printer on the host machine's local area network.
- If your host host machine has access to the Internet, cloud-based printing is supported through cloud-printing services such as Google Cloud Print and HP Mobile Printing.

USB Device Support

Connecting USB devices to the Graphics Agent for macOS is not supported. Keyboards, mice, and other pointer devices are managed by PCoIP clients, and are always allowed.

Relative mouse movements, a feature used in many gaming applications or technologies built on Unity, is not supported.

Wacom Support

The Graphics Agent for macOS supports *locally terminated* Wacom tablets, where peripheral data is processed locally at the PCoIP Client.

Locally terminated Wacom tablets are much more responsive and tolerate high-latency connections, but may not support advanced features like Touch. For information regarding feature support and PCoIP client and agent requirements, see the following sections.

Locally Terminated Wacom Tablets

Locally-terminated tablets have greatly improved responsiveness, and tolerate higher-latency (including 25ms and higher) networks.

For the best experience and most complete device support, use the latest available PCoIP agent, PCoIP software client, and PCoIP Zero Client firmware. To find out when support was added for individual Wacom device, refer to the release notes for your client.

The following Wacom tablet models have been tested and are supported with local termination mode:

PCoIP client support for *locally terminated* Wacom tablets and the Graphics Agent for macOS

	PCoIP Tera2 Zero Client *6.2.0+, except as noted*	PCoIP Software Client for Windows	PCoIP Software Client for macOS	PCoIP Software Client for Linux
Intuos Pro Medium *PTH-660*	✔	✔	✔	✔
Intuos Pro Large *PTH-860*	✔	✔	✔	✔

Important: Touch is not supported

Touch features of Wacom devices are not supported with local termination.

Other Wacom tablets may work, but have not been tested and should not be used in production environments.

Installing the Graphics Agent for macOS

Always have a secondary means of access to your remote machine

Currently, the Graphics Agent for macOS can be impacted by operating system-level events which may cause PCoIP connections to stop working. To recover from these situations, you must be able to access the remote machine directly, either physically or via VNC. See Troubleshooting for help.

Before starting, be aware of the following conditions:

- Firewall settings

  The macOS machine that will act as the PCoIP host must allow PCoIP traffic through its firewall. Firewall settings should be either:

  - The firewall is *off*, or
  - The firewall is *on*, and:
    - Automatically allow downloaded signed software to receive incoming connections is checked, *or*
    - Automatically allow downloaded signed software to receive incoming connections is *not* checked, *and* the following executables are allowed through the firewall:
      - `/Applications/PCoIP Agent.app`
      - `/Applications/PCoIP Agent.app/Contents/MacOS/pcoip-agent`
      - `/Applications/PCoIP Agent.app/Contents/MacOS/pcoip-server`

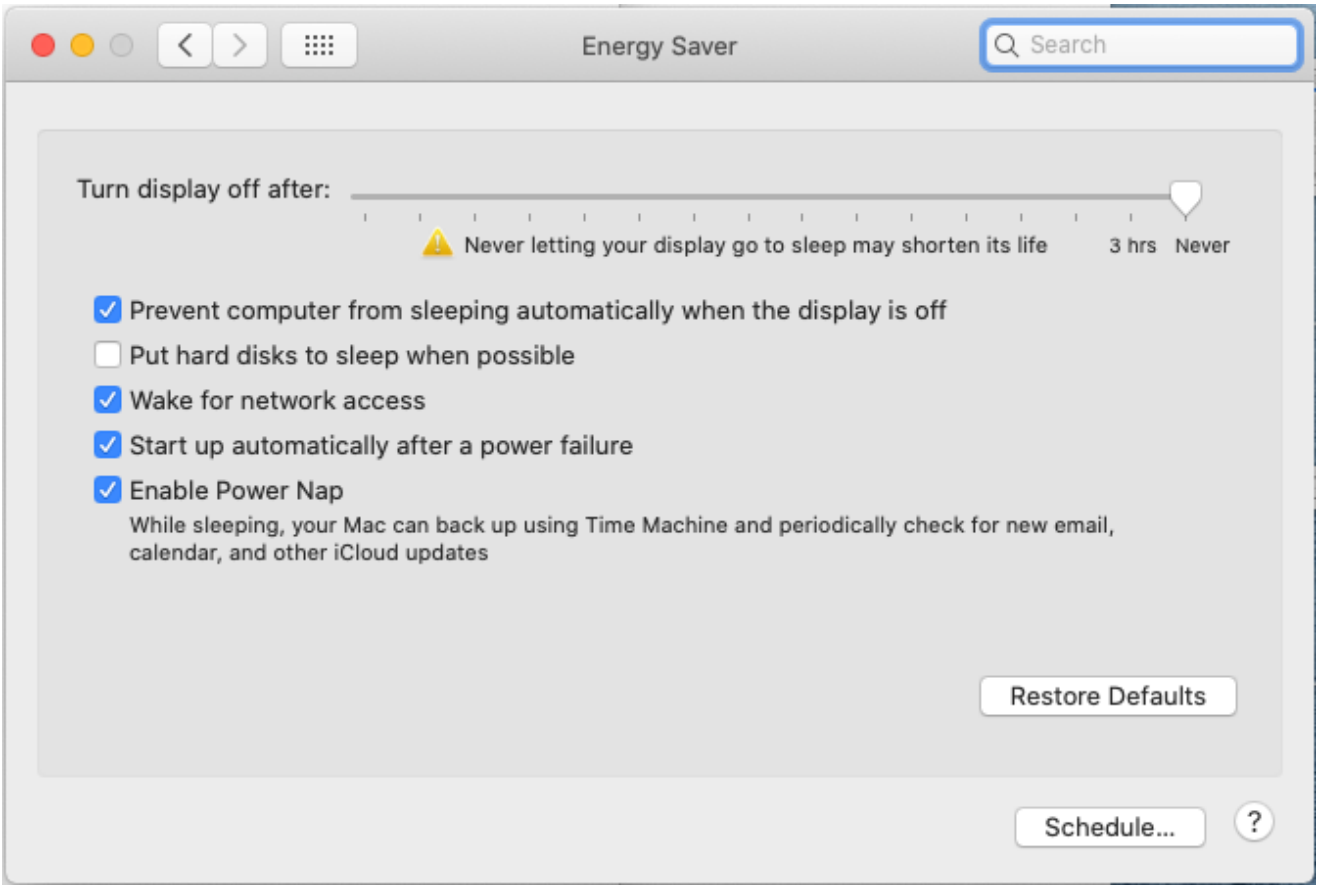
    If the firewall is on and *Automatically allow downloaded signed software to receive incoming connections* is not checked, you will be prompted to allow the above permissions during installation.

    Finding *Automatically allow downloaded signed software to receive incoming connections*

    The *Automatically allow downloaded signed software to receive incoming connections* setting is in System Preferences > Security & Privacy, under the Firewall tab. Click Firewall Options to reveal it. The *Firewall Options* button is only visible if the firewall is on.

- Energy Saver settings

Energy Saver features can cause the remote system to go to sleep or become unresponsive. To prevent this, open **System Preferences** > **Energy Saver**, and configure the settings as follows:



Setting	Value
Turn off display after	set slider to *Never*
Prevent computer from sleeping automatically when the display is off	*checked*
Put hard disks to sleep when possible	*unchecked*
Wake for network access	*checked*
Start up automatically after power failure	*checked*
Enable Power Nap	*checked*

- **Create a user account for PCoIP Connections**. The user name cannot contain spaces, and cannot be the ***root user*** account (the root user is an administrative account with elevated permissions, and is disabled by default in macOS).

Installing With the Interface

1. Download the installer to the machine you'll be using as the PCoIP host.
2. Run the installer **`pkg`**.
3. Click through the installer steps and accept the End User License Agreement. The agent application will be installed.
4. Open the PCoIP Agent application, found in **`/System/Volumes/Data/Applications/PCoIP Agent.app`**.
5. You will be prompted to allow **Accessibility** permission. Grant and confirm privacy permissions for Accessibility.
6. Open the PCoIP Agent application again.
7. You will be prompted to allow **Screen Recording** permission. Grant and confirm privacy permissions for Screen Recording.

> **Note: No user interface**
>
> The PCoIP Agent application launches PCoIP services which continue to run in the background. There is no user interface and no further direct interaction with the application. You do not need to launch **`PCoIP Agent.app`** again after installation.

8. Next, provide the license registration code you received from Teradici.

> **Important: This step does not apply for PCoIP License Server users**
>
> If you are using a local PCoIP License Server, do not follow this step; refer to Licensing the Graphics Agent for macOS.

If you are using Teradici Cloud Licensing, open a terminal window and type one of the following commands, depending on whether your host machine is behind a proxy server:

- If you use a proxy server to access the internet, provide this command (replace `<XXXXXX@YYY-YYYY-YYY>`, `<PROXY_SERVER_ADDRESS>`, and `<PROXY_PORT_NUMBER>` with your own values):

```
sudo pcoip-register-host --registration-code=<XXXXXX@YYY-YYYY-YYY> --
proxy-server=<PROXY_SERVER_ADDRESS> --proxy-port=<PROXY_PORT_NUMBER>
```

- If your system does *not* use a proxy server, provide this command (replace `{REGISTRATION_CODE}` with your own value):

```
sudo pcoip-register-host --registration-code=<XXXXXX@YYY-YYYY-YYY>
```

If you are using a PCoIP License Server, refer to Licensing the Graphics Agent for macOS for licensing instructions.

9. Restart the machine.

Installing With the Terminal

1. Download the installer to the machine you'll be using as the PCoIP host.
2. Open a terminal window.
3. Run the installer using the following command. Replace {PACKAGE_NAME} with the name of the `.pkg` file you downloaded:

```
sudo installer -pkg {PACKAGE_NAME}.pkg -target /
```

   The installer will return a `0` if successful. A return value of anything other than `0` indicates a failure.

4. Open the PCoIP Agent application, found in `/System/Volumes/Data/Applications/PCoIP Agent.app`.
5. You will be prompted to allow **Accessibility** permission. Grant and confirm privacy permissions for Accessibility.
6. Open the PCoIP Agent application again.

7. You will be prompted to allow **Screen Recording** permission. Grant and confirm privacy permissions for Screen Recording.

   > **Note: No user interface**
   >
   > The PCoIP Agent application launches PCoIP services which continue to run in the background. There is no user interface and no further direct interaction with the application. You do not need to launch `PCoIP Agent.app` again after installation.

8. Next, provide the license registration code you received from Teradici.

   > **Important: This step does not apply for PCoIP License Server users**
   >
   > If you are using a local PCoIP License Server, do not follow this step; refer to Licensing the Graphics Agent for macOS.

   If you are using Teradici Cloud Licensing, type one of the following commands, depending on whether your host machine is behind a proxy server:

   - If you use a proxy server to access the internet, provide this command (replace `<XXXXXX@YYY-YYYY-YYY>`, `<PROXY_SERVER_ADDRESS>`, and `<PROXY_PORT_NUMBER>` with your own values):

     ```
     sudo pcoip-register-host --registration-code=<XXXXXX@YYY-YYYY-YYY> --
     proxy-server=<PROXY_SERVER_ADDRESS> --proxy-port=<PROXY_PORT_NUMBER>
     ```

   - If your system does *not* use a proxy server, provide this command (replace `<XXXXXX@YYY-YYYY-YYY>` with your own registration code value):

     ```
     sudo pcoip-register-host --registration-code={REGISTRATION_CODE}
     ```

   If you are using a PCoIP License Server, refer to Licensing the Graphics Agent for macOS for licensing instructions.

9. Restart the machine.

Licensing The Graphics Agent for macOS

The Graphics Agent for macOS must be assigned a valid PCoIP session license before it will work. Until you've registered it, you can't connect to the desktop using a PCoIP client.

You receive a registration code when you purchase a pool of licenses from Teradici. Each registration code can be used multiple times; each use consumes one license in its pool.

> **Note: Registration code format**
>
> Registration codes look like this: `ABCDEFGH12@AB12-C345-D67E-89FG`

PCoIP agent license registrations are managed automatically by Teradici's Cloud Licensing service. If necessary, you can manage them yourself, using your own locally-installed PCoIP license server instead.

If you need to purchase licenses, contact Teradici.

Troubleshooting Licensing Issues

If you're encountering problems with Teradici licensing, refer to Troubleshooting License Issues.

Using Teradici Cloud Licensing

To use Cloud Licensing, all you need to do is provide a registration code for each PCoIP agent in your deployment (the same registration code can be used multiple times).

To provide the registration code:

SSH into the agent machine, and invoke `pcoip-register-host` with the license registration code and proxy settings if required:

```
sudo pcoip-register-host --registration-code=<XXXXXX@YYY-YYYY-YYY> [--proxy-server=<proxy-server-address>] [--proxy-port=<proxy-port-number>]
```

Whitelist network blocks for Teradici Cloud Licensing

If you are using Teradici Cloud Licensing, you will need to whitelist the following:

- teradici.flexnetoperations.com
- teradici.compliance.flexnetoperations.com

Alternatively, you can also ensure the following network blocks are whitelisted:

- **Production**: 64.14.29.0/24
- **Disaster Recovery**: 64.27.162.0/24

The following network blocks are not currently in use, but may also be used in the future:

- **Production**: 162.244.220.0/24
- **Disaster Recovery**: 162.244.222.0/24

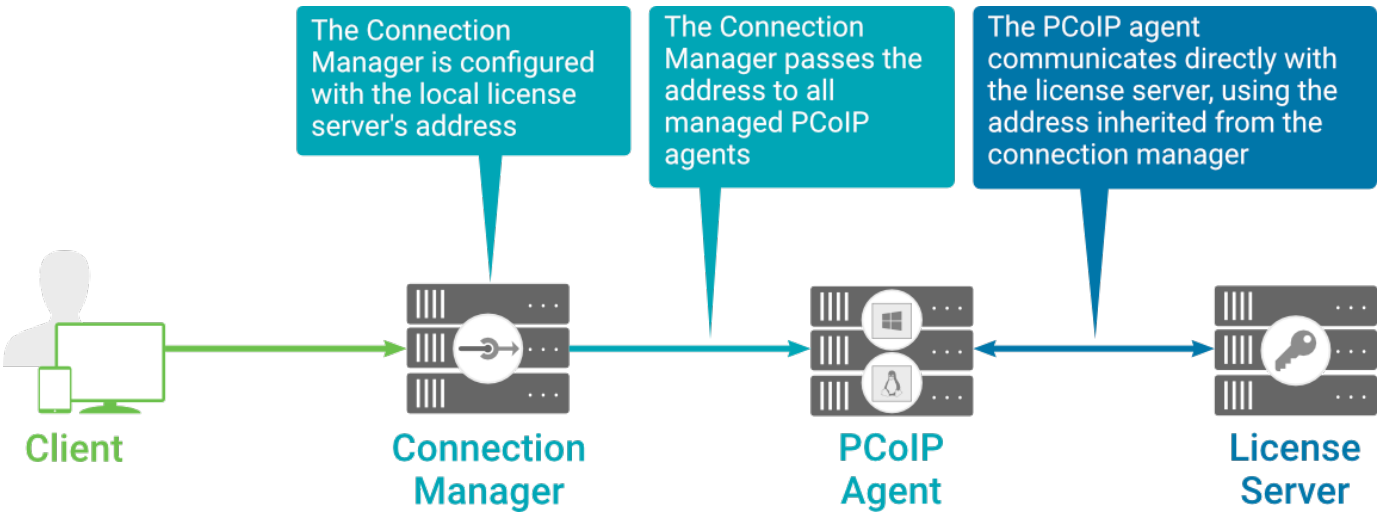
Licensing PCoIP Agents With a Local License Server

In deployments where PCoIP agents cannot access the internet, or where cloud-based licensing is not permitted or desired, a local PCoIP License Server can be used instead. The PCoIP License Server manages PCoIP session licenses within your private environment.

Configuring PCoIP agents to use a local license server is done in one of two ways, depending on whether your deployment uses a PCoIP Connection Manager, or whether your PCoIP clients connect directly to PCoIP agents.

Brokered Environment Licensing

In ***brokered*** deployments, the license server address is configured in the Connection Manager, which passes it through to its managed PCoIP agents.



Local license validation using a Graphics Agent for macOS and a brokered connection

When using a Connection Manager, the license server address is only configured once no matter how many PCoIP agents are behind the Connection Manager.

To set the License Server URL in the Connection Manager:

1. On the Connection Manager machine, use a text editor to open /etc/ConnectionManager.conf.
2. Set the `LicenseServerAddress` parameter with the address of your local license server:
   - `http://{license-server-address}:{port}/request`
3. Save and close the configuration file.
4. Restart the Connection Manager.

Verifying Your Brokered Licensing Configuration

To verify your system's licensing configuration, run `pcoip-validate-license` from the console on the Graphics Agent for macOS machine. The command will ping the license server and attempt to retrieve information on an available license:

```
sudo pcoip-validate-license --license-server-url <license-server-address> [
--proxy-server <proxy-server-address>] [--proxy-port <proxy-port-number]
```

Where `<license-server-address>` is the address of the license server to ping, formatted as `http://{license-server-address}:{port}/request`

If the license server is behind a proxy server, provide the proxy information via the `--proxy-server` and `--proxy-port` parameters.

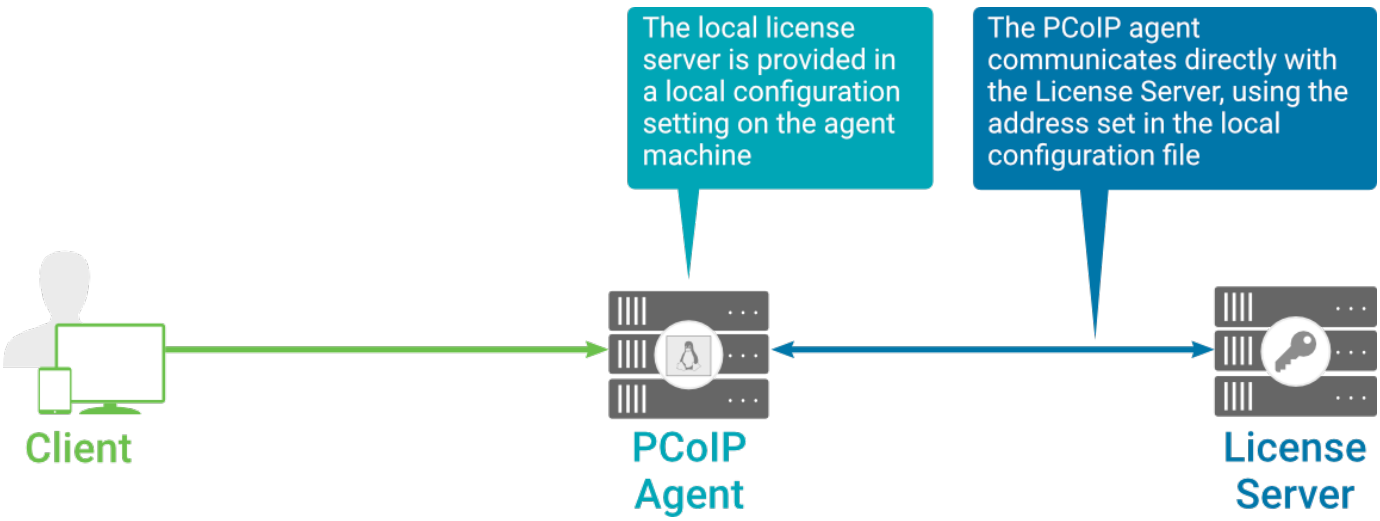
If successful, the response will show that a valid license was found on the license server, and its expiration date.

If the connection is unsuccessful, investigate the following possibilities:

- The license server address is incorrect, or formatted incorrectly.
- The license server is inaccessible.
- There are no available licenses on the license server. `pcoip-validate-license` will only return a positive response if there is at least one available session license.
- If you have only one license on the license server and run `pcoip-validate-license` from a PCoIP session, the command will fail because you are currently using the single license. In this scenario, disconnect your PCoIP session and try again from an SSH session instead.

Unbrokered Environment Licensing

In direct, or unbrokered, deployments, each PCoIP agent is configured with the license server address via a local agent setting. When a client initiates a new PCoIP session, the PCoIP agent uses its local configuration to communicate with the license server.



Local license validation using a Graphics Agent for macOS and a direct (unbrokered) connection

Each PCoIP agent in your environment must be individually configured with the license server's URL.

To configure the License Server URL on the Graphics Agent for macOS machine:

1. Add or modify the `pcoip.license_server_path` directive.

   In this command, replace `{LICENSE_SERVER_PATH}` and `{LICENSE_SERVER_PORT}` with your own values:

   ```
   sudo defaults write "/Library/Preferences/com.teradici.pcoip-agent.plist"
   pcoip.license_server_path "https://{LICENSE_SERVER_PATH}:
   {LICENSE_SERVER_PORT}/request";
   ```

2. If the license server is behind a proxy server, provide the proxy information.

   In this command, replace `{PROXY_SERVER_PATH}` and `{PROXY_PORT}` with your own values:

   ```
   sudo defaults write "/Library/Preferences/com.teradici.pcoip-agent.plist"
   pcoip.license_proxy_server {PROXY_SERVER_PATH}
   sudo defaults write "/Library/Preferences/com.teradici.pcoip-agent.plist"
   pcoip.license_proxy port {PROXY_PORT};
   ```

The changes will take effect on the next PCoIP session.

Verifying Your Unbrokered Licensing Configuration

To verify your system's licensing configuration, run `pcoip-validate-license` from the console on the Graphics Agent for macOS machine. The command will ping the license server and attempt to retrieve information on an available license:

```
sudo pcoip-validate-license --license-server-url <license-server-address> [
--proxy-server <proxy-server-address>] [--proxy-port <proxy-port-number]
```

Where `<license-server-address>` is the address of the license server to ping, formatted as `http://{license-server-address}:{port}/request`

If the license server is behind a proxy server, provide the proxy information via the `--proxy-server` and `--proxy-port` parameters.

If successful, the response will show that a valid license was found on the license server, and its expiration date.

If the connection is unsuccessful, investigate the following possibilities:

- The license server address is incorrect, or formatted incorrectly.
- The license server is inaccessible.
- There are no available licenses on the license server. `pcoip-validate-license` will only return a positive response if there is at least one available session license.
- If you have only one license on the license server and run `pcoip-validate-license` from a PCoIP session, the command will fail because you are currently using the single license. In this scenario, disconnect your PCoIP session and try again from an SSH session instead.

Updating the Graphics Agent for macOS

To update the Graphics Agent for macOS to a new version, obtain the new installer and run it in place, replacing the older version. Your configuration settings will be preserved.

Uninstalling the Graphics Agent for macOS

To uninstall the Graphics Agent for macOS, drag the **`PCoIP Agent.app`** application to the trash.

Alternatively, you can run the installer directly:

```
sudo /Library/Application\ Support/Teradici/pcoip/uninstaller
```

Configuring the Graphics Agent for macOS

You can configure the PCoIP agent, and optimize PCoIP protocol behavior for local network conditions, by adjusting configuration directives found in **`/Library/Preferences/com.teradici.pcoip-agent.plist`**.

You can find detailed information and descriptions about each setting in the next section. You can also consult the `man` pages for **`com.teradici.pcoip-agent.plist`**:

```
man com.teradici.pcoip-agent.plist
```

Viewing Current Settings

To view the contents of your current settings file, open a Terminal window and run the following command:

```
defaults read "/Library/Preferences/com.teradici.pcoip-agent.plist";
```

The **`com.teradici.pcoip-agent.plist`** file only holds configuration overrides; it does not hold all configuration values. Values not specified in the file will use their defaults, as documented in the next section.

The configuration file will not exist until at least one setting has been written to it. If you try to read the file before it exists, you will receive a `Domain com.teradici.pcoip-agent.plist does not exist` error.

Applying Configuration Changes

To set or change a configuration value, use `defaults write` to modify directives in **`com.teradici.pcoip-agent.plist`**. The format is:

```
sudo defaults write /Library/Preferences/com.teradici.pcoip-agent.plist
{directive} {value};
```

...where `{directive}` is one of the settings detailed in the next section, and `{value}` is the desired setting.

For example, to set the maximum bandwidth to ***900000 kilobits/second***, and the device bandwidth floor to ***5000 kilobits/second***, you would set values like this:

```
sudo defaults write /Library/Preferences/com.teradici.pcoip-agent.plist
pcoip.max_link_rate 900000;
sudo defaults write /Library/Preferences/com.teradici.pcoip-agent.plist
pcoip.device_bandwidth_floor 5000;
```

A complete list of configurable values is shown next.

Configurable Settings

The following settings can be configured on the Graphics Agent for macOS. Refer to Configuring the PCoIP agent to understand how to modify these settings.

Build-to-lossless

Directive	Options	Default
`pcoip.enable_build_to_lossless`	**0** (off), **1** (on)	Off

This setting takes effect immediately. Specifies whether to turn the build-to-lossless feature of the PCoIP protocol off or on; this feature is turned off by default.

When build-to-lossless is turned off images and other desktop content may never build to a lossless state. In network environments with constrained bandwidth, turning off build-to-lossless can provide bandwidth savings. Build-to-lossless is recommended for environments that require images and desktop content to be built to a lossless state.

Clipboard redirection

Directive	Options	Default
`pcoip.server_clipboard_state`	0—Disabled in both directions 1—Enabled in both directions 2—Enabled client to agent only 3—Enabled agent to client only	—

This setting takes effect when you start the next session. Determines the direction in which clipboard redirection is allowed. You can select one of these values:

- Disabled in both directions
- Enabled in both directions (default setting)
- Enabled client to agent only (That is, allow copy and paste only from the client system to the host desktop.)
- Enabled agent to client only (That is, allow copy and paste only from the host desktop to the client system.)

Clipboard redirection is implemented as a virtual channel. If virtual channels are disabled, clipboard redirection does not function.

Connection addresses

Directive	Options	Default
`pcoip.connection_address`	string *(up to 511 characters)*	—
`pcoip.client_connection_address`	string *(up to 511 characters)*	—

This setting takes effect when you start the next session. Configuring this allows you to control the IPv4 or IPv6 address used by the agent or client in PCoIP sessions.

'Connection Address' controls the IP address used by the agent for the PCoIP session.

'Client Connection Address' controls the IP address the client is told to use when establishing the PCoIP session.

Please note that neither of these values should need to be set under normal circumstances.

Enable lock screen when a client connects or disconnects

Directive	Options	Default
`pcoip.enable_screen_lock`	**0** (off), **1** (on)	On

This setting determines whether or not the screen is locked during a client connection, and upon client disconnection. When enabled, workstations or baremetal PCoIP Graphics Agents screen will be locked. Enable this setting to enforce privacy on workstations or baremetal PCoIP Graphics Agents.

Enable/disable audio in the PCoIP session

Directive	Options	Default
`pcoip.enable_audio`	**0** (off), **1** (on)	On

This setting takes effect when you start the next session. Determines whether audio is enabled in PCoIP sessions. Both endpoints must have audio enabled. When this setting is enabled, PCoIP audio is allowed. When it is disabled, PCoIP audio is disabled. When this setting is not configured, audio is enabled by default.

Hide local cursor

Directive	Options	Default
`pcoip.disable_locally_rendered_cursor`	**0** (off), **1** (on)	Off

This setting takes effect immediately. When this setting is enabled the local cursor on the client will be hidden. This may resolve duplicate cursor issues if there is a host rendered cursor within the host environment but may also result in no visible cursor. With this setting enabled there may be delays in mouse movements due to network latency and video processing times. By default, this setting is disabled, meaning that local cursors will be used, providing the most responsive user experience.

License server URL

Directive	Options	Default
`pcoip.license_server_path`	string *(up to 511 characters)*	—

This setting takes effect when you start the next session. This policy sets the license server path. Enter the license server path in 'https://address:port/request' or 'http://address:port/request' format.

Maximum PCoIP session bandwidth

Directive	Range	Increment	Default
`pcoip.max_link_rate`	104 – 900000	100	900000

This setting takes effect when you start the next session. Specifies the maximum bandwidth, in kilobits per second, in a PCoIP session. The bandwidth includes all imaging, audio, virtual channel, USB, and control PCoIP traffic.

Set this value based on the overall capacity of the link to which your endpoint is connected, taking into consideration the number of expected concurrent PCoIP sessions. For example, with a single user VDI configuration (e.g. a single PCoIP session) that connects through a 4Mbit/s Internet connection, set this value to 4Mbit (or 10% less than this value to leave some allowance for other network traffic).

Setting this value prevents the agent from attempting to transmit at a higher rate than the link capacity, which would cause excessive packet loss and a poorer user experience. This value is

symmetric. It forces the client and agent to use the lower of the two values that are set on the client and agent side. For example, setting a 4Mbit/s maximum bandwidth forces the agent to transmit at a lower rate, even though the setting is configured on the client.

When this setting is disabled or not configured on an endpoint, the endpoint imposes no bandwidth constraints. When this setting is configured, the setting is used as the endpoint's maximum bandwidth constraint in kilobits per second.

The default value when this setting is not configured is 900000 kilobits per second.

This setting applies to the agent and client. If the two endpoints have different settings, the lower value is used.

PCoIP Security Settings

Directive	Options	Default
`pcoip.tls_security_mode`	0—Maximum Compatibility	—

This setting takes effect when you start the next session. Controls the cryptographic cipher suites and encryption ciphers used by PCoIP endpoints.

The endpoints negotiate the actual cryptographic cipher suites and encryption ciphers based on the settings configured here. Newer versions of TLS and stronger cipher suites will be preferred during negotiation between endpoints.

If this setting is not configured or disabled, the TLS Security Mode will be set to Maximum Compatibility, and the PCoIP Data Encryption Ciphers will be set to AES-256-GCM, AES-128-GCM.

TLS Security Mode

Maximum Compatibility offers TLS 1.2 and TLS 1.3, and a range of cipher suites including those that support Perfect Forward Secrecy (PFS) and SHA-1. Supported cipher suites:

- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256

- TLS_RSA_WITH_AES_256_GCM_SHA384
- TLS_RSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA

Blacklisted Cipher Suites

Provides the ability to block specific cipher suites from being offered during negotiation. Must be entered as a semi-colon separated list of cipher suites.

PCoIP Data Encryption Ciphers

Encryption ciphers used for PCoIP UDP data encryption. "AES-256-GCM, AES-128-GCM" is the default setting. AES-256-GCM will get negotiated if the client supports it, otherwise, AES-128-GCM will get negotiated.

PCoIP Ultra

Directive	Options	Range	Increment	Default
`pcoip.ultra`	0—Disabled 1—CPU Offload 2—GPU Offload 3—Automatic Offload			—
`pcoip.ultra_offload_mpps`		1 – 40	1	10

This setting takes effect when you start the next session. When this setting is disabled or not configured then PCoIP Ultra will not be used.

- PCoIP Ultra CPU Offload - these optimizations require CPU support for the AVX2 instruction set on both the remote host and client and are not compatible with the PCoIP Zero client. CPU Offload is recommended for 4K UHD resolutions with video playback requirements of 30 fps (or more) and highest image quality / color accuracy.

- PCoIP Ultra GPU Offload - these optimizations require an NVIDIA graphics card on the remote host capable of NVENC. GPU Offload is recommended when minimal CPU impact of pixel encoding is desired.
- PCoIP Ultra Auto Offload - enabling this setting allows PCoIP to automatically switch between CPU and GPU Offload modes; CPU Offload is used by default to provide the best image fidelity, GPU Offload is used during periods of high display activity to provide improved frame rates and bandwidth optimization. This setting is only effective if the remote host and client endpoints are capable of both CPU and GPU Offload.

The PCoIP Ultra Offload MPPS sets the Megapixels Per Second (MPPS) transition rate between PCoIP Ultra CPU Offload and PCoIP Ultra GPU Offload. Under Auto-Offload, PCoIP Ultra uses CPU Offload at lower pixel rates and switches to GPU Offload at the Offload MPPS. Increasing this value results in PCoIP Ultra transitioning to GPU Offload at a higher pixel rate and decreasing this value results in the transition at a lower pixel rate. The default PCoIP Ultra Offload MPPS is set to 10.

PCoIP event log verbosity

Directive	Range	Increment	Default
`pcoip.event_filter_mode`	0 – 3	1	2

This setting takes effect immediately. Configures the PCoIP event log verbosity ranging from 0 (least verbose) to 3 (most verbose).

PCoIP image quality levels

Directive	Options	Range	Increment	Default
`pcoip.minimum_image_quality`		30 – 100	10	40
`pcoip.maximum_initial_image_quality`		30 – 100	10	80
`pcoip.frame_rate_vs_quality_factor`		0 – 100	10	50
`pcoip.maximum_frame_rate`		0 – 60	1	—

Directive	Options	Range	Increment	Default
`pcoip.use_client_img_settings`	**0** (off), **1** (on)			Off

This setting takes effect immediately. Controls how PCoIP renders images during periods of network congestion. The Minimum Image Quality, Maximum Initial Image Quality, and Maximum Frame Rate values interoperate to provide fine control in network-bandwidth constrained environments.

Use the Minimum Image Quality value to balance image quality and frame rate for limited-bandwidth scenarios. You can specify a value between 30 and 100. The default value is 40. A lower value allows higher frame-rates, but with a potentially lower quality display. A higher value provides higher image quality, but with potentially lower frame rates when network bandwidth is constrained. When network bandwidth is not constrained, PCoIP maintains maximum quality regardless of this value.

Use the Maximum Initial Image Quality value to reduce the network bandwidth peaks required by PCoIP by limiting the initial quality of the changed regions of the display image. You can specify a value between 30 and 100. The default value is 80. A lower value reduces the image quality of content changes and decreases peak bandwidth requirements. A higher value increases the image quality of content changes and increases peak bandwidth requirements. Unchanged regions of the image progressively build to a lossless (perfect) quality regardless of this value. A value of 80 or lower best utilizes the available bandwidth.

The Minimum Image Quality value cannot exceed the Maximum Initial Image Quality value.

Use the Frame Rate vs Image Quality value to favor image sharpness over smooth motion during a PCoIP session when network bandwidth is limited. Lower values favor smoothness, higher values favor sharpness of image.

Use the Maximum Frame Rate value to manage the average bandwidth consumed per user by limiting the number of screen updates per second. You can specify a value between 1 and 60 frames per second. A higher value can use more bandwidth but provides less jitter, which allows smoother transitions in changing images such as video. A lower value uses less bandwidth but results in more jitter.

YUV chroma subsampling is set to 4:4:4 by default for maximum image quality. Setting YUV chroma subsampling to 4:2:0 is only supported in combination with PCoIP Ultra GPU optimization.

This setting will enable chroma subsampling to further compress the imaging to reduce bandwidth usage at the cost of reduced color accuracy. Please note: 4:4:4 subsampling with PCoIP Ultra GPU optimization is GPU dependent and is not supported by all GPUs, in this case PCoIP will fallback to 4:2:0 subsampling. Please see our support site for further details.

Set the 'Use image settings from zero client' when you want to use the 'Minimum Image Quality', 'Maximum Initial Image Quality', 'Maximum Frame Rate', 'Disable Build to Lossless' values from the client instead of the host. Currently, only Zero Client Firmware 3.5 and above support these settings on the client side.

These image quality values apply to the soft host only and have no effect on a soft client.

When this setting is disabled or not configured, the default values are used.

PCoIP session MTU

Directive	Range	Increment	Default
`pcoip.mtu_size`	500 – 1500	1	1200

This setting takes effect when you start the next session. Specifies the Maximum Transmission Unit (MTU) size for UDP packets for a PCoIP session.

The MTU size includes IP and UDP packet headers. TCP uses the standard MTU discovery mechanism to set MTU and is not affected by this setting. The maximum MTU size is 1500 bytes. The minimum MTU size is 500 bytes. The default value is 1200 bytes.

Typically, you do not have to change the MTU size. Change this value if you have an unusual network setup that causes PCoIP packet fragmentation.

This setting applies to the agent and client. If the two endpoints have different MTU size settings, the lowest size is used.

If this setting is disabled or not configured, the client uses the default value in the negotiation with the agent.

PCoIP session audio bandwidth limit

Directive	Range	Increment	Default
`pcoip.audio_bandwidth_limit`	0 – 100000	1	512

This setting takes effect immediately. Specifies the maximum audio bandwidth that can be used for audio output (sound playback) from the virtual desktop to the client in a PCoIP session. Note that the network transport overhead can add an additional 20-40% bandwidth to this number.

Audio processing monitors the bandwidth needed for audio and selects the audio compression algorithm that provides the best quality possible, without exceeding the bandwidth limit:

- 512 kbit/s or higher - 7.1 surround, high-quality, compressed audio
- 384 kbit/s or higher - 5.1 surround, high-quality, compressed audio
- 256 kbit/s or higher - stereo, high-quality, compressed audio
- 48 kbit/s to 255 kbit/s - stereo audio ranging between FM radio quality down to AM radio quality
- 32 kbit/s to 47 kbit/s - monaural AM radio or phone call quality
- Below 32 kbit/s - results in no audio playback

If this setting is not configured, a default audio bandwidth limit of 512 kbit/s is configured to constrain the audio compression algorithm selected.

Note that zero clients on older firmware have less efficient audio compression algorithms that may require setting this limit higher to achieve the same audio quality or upgrading the firmware.

PCoIP session bandwidth floor

Directive	Range	Increment	Default
`pcoip.device_bandwidth_floor`	0 – 100000	1	—

This setting takes effect immediately. Specifies a lower limit, in kilobits per second, for the bandwidth that is reserved by the PCoIP session.

This setting configures the minimum expected bandwidth transmission rate for the endpoint. When you use this setting to reserve bandwidth for an endpoint, the session does not have to wait for bandwidth to become available, which improves session responsiveness.

Make sure that you do not over-subscribe the total reserved bandwidth for all endpoints. Make sure that the sum of bandwidth floors for all connections in your configuration does not exceed the network capability.

The default value is 0, which means that no minimum bandwidth is reserved. When this setting is disabled or not configured, no minimum bandwidth is reserved.

This setting applies to the agent and client, but the setting only affects the endpoint on which it is configured.

PCoIP statistics interval

Directive	Range	Increment	Default
`pcoip.server_statistics_interval_seconds`	0 – 65535	1	—

This setting takes effect immediately. Configuring this allows you to set an interval in seconds for logging performance statistics to the PCoIP server log. When not configured, logging is disabled by default.

PCoIP transport header

Directive	Options	Default
`pcoip.transport_session_priority`	1—High Priority 2—Medium Priority (default) 3—Low Priority 4—Undefined Priority	—

This setting takes effect when you start the next session. Configures the PCoIP transport header.

PCoIP transport header is a 32-bit long header which is added to all PCoIP UDP packets (only if the transport header is enabled/supported by both sides). PCoIP transport header allows network devices to make better prioritization/Qos decisions when dealing with network congestions. The transport header is enabled by default.

The transport session priority determines the PCoIP session priority reported in the PCoIP Transport Header. Network devices make better prioritization/Qos decisions based on the specified transport session priority. The transport session priority value is negotiated by the PCoIP agent and client. If agent has specified a transport session priority value (high, medium, or low), then the session uses the agent specified session priority. If only the client has specified a transport session priority (high, medium, or low), then the session uses the client specified session priority. If neither agent nor client has specified a transport session priority (or specified 'undefined priority'), then the session uses/defaults to the medium session priority.

PCoIP virtual channels

Directive	Options	Default
`pcoip.vchan_list`	string *(up to 255 characters)*	—

This setting takes effect when you start the next session. Specifies the virtual channels that can or cannot operate over a PCoIP session.

There are two modes of operation:

- Enable all virtual channels except for <list> (default setting)
- Disable all virtual channels except for <list>

When specifying which virtual channels to include or not include in the list, the following rules apply:

- An empty list is allowed
- Multiple virtual channel names in the list must be separated by the vertical bar (|) character. For example: channelA|channelB

- Vertical bar or backslash () characters in virtual channel names must be preceded by a backslash. For example: the channel name "awk|ward\channel" must be specified as "awk|ward\channel" (without the double quotes)
- A maximum of 15 virtual channels are allowed in a single PCoIP session

The virtual channel must be enabled on both agent and client for it to be used.

Proxy Access to a remote License Server

Directive	Options	Range	Increment	Default
`pcoip.license_proxy_server`	string *(up to 511 characters)*			—
`pcoip.license_proxy_port`		0 – 65535	1	—

This setting takes effect when you start the next session. If a proxy is required to access a local License Server or the Cloud License Server, enter those parameters here. These parameters are loaded only during agent startup.

Timezone redirection

Directive	Options	Default
`pcoip.enable_timezone_redirect`	**0** (off), **1** (on)	On

This setting takes effect when you start the next session. Configuring this allows you to enable or disable timezone redirection. When not configured, timezone redirection is enabled by default.

User collaboration

Directive	Options	Range	Increment	Default
`pcoip.enable_collaboration`	**0** (off), **1** (on)			Off
`pcoip.collaboration_udpport`		1 – 65535	1	64172

This setting takes effect when the agent is restarted. This policy enables or disables user collaboration. When not configured, user collaboration is disabled by default.

The default UDP port used for collaborator sessions is 64172. When a different port is used, ensure that firewall rules are adjusted so that PCoIP traffic can go through the new port.

Making a Connection from a PCoIP Client

Important: Accept login dialogs first

macOS login dialogs will prevent PCoIP client connections from succeeding. Before you attempt to connect a PCoIP session, you must log in to the remote computer, using the account that will accept PCoIP connections, either at the physical machine or via VNC or screen sharing.

After dismissing all log in dialogs you may then log off, and connect remotely via PCoIP.

Connection troubleshooting

If you are unable to connect, we have troubleshooting information that can help. See Troubleshooting Connection Issues.

Once you've installed and configured your Graphics Agent for macOS, you're ready to accept incoming connections from remote ***PCoIP Clients***. PCoIP clients are remote endpoint devices available in as software or firmware and make secure PCoIP connections to the remote desktop through the installed Graphics Agent for macOS.

For more information about PCoIP client connectivity requirements and usage instructions, see the following documentation:

- Software clients:
  - Teradici PCoIP Software Client for Windows
  - Teradici PCoIP Software Client for macOS
  - Teradici PCoIP Software Client for Linux
- Mobile Clients:
  - Teradici PCoIP Mobile Client for iOS
  - Teradici PCoIP Mobile Client for Android
  - Teradici PCoIP Mobile Client for Chromebooks
- Zero clients:
  - Teradici Tera2 PCoIP Zero Client

PCoIP Agent Deployment and Client Connectivity Requirements

PCoIP clients can connect to your desktops hosted in proof-of-concept, cloud, or datacenter deployments. Requirements and network security levels will vary depending on your deployment type. See Supported PCoIP Architectures for each deployment's components and requirements.

Managing Client Connections

In most cases, PCoIP clients connect to PCoIP agents through a ***connection broker***. The broker is responsible for matching users to their available desktops, and then establishing the PCoIP session with their selected resource.

PCoIP agents do not need to be configured to use these brokering services. All relevant configuration is done at the broker, which then communicates with the agent.

Brokering Options

There are several ways you can manage client connections to remote desktops

Direct Connections

In direct connection scenarios—where a broker is not involved—the PCoIP agent acts as its own broker. In these cases, a client user will provide the IP address or FQDN of the agent machine to their client, and the connection is made securely with no intermediate step.

Teradici Cloud Access Manager

Teradici Cloud Access Manager is a cloud-based service available as part of Cloud Access Software that centrally manages PCoIP deployments. It enables highly scalable and cost-effective Cloud Access Software deployments by managing cloud compute costs and brokering PCoIP connections to remote Windows or Linux workstations.

Teradici PCoIP Connection Manager

The **Teradici PCoIP Connection Manager** is provided in a bundle with the **Teradici PCoIP Security Gateway**, and allows self-managed brokering services. For information about the Teradici PCoIP Connection Manager, including installation and configuration instructions, see the Connection Manager and Security Gateway documentation.

Third-party Connection Brokers

Teradici PCoIP agents also support third-party connection brokers. For a current list of brokering partners, see Teradici Technology Partners on Teradici's website.

Security Certificates in PCoIP Agents

PCoIP requires a certificate to establish a session. By default, PCoIP agents generate a self-signed certificate that secures the PCoIP session. Each component in the PCoIP system can generate these self-signed certificates, which will automatically work together without requiring any configuration.

You can, if needed, create and deploy your own custom certificates instead of relying on Teradici's self-signed certificates. This section explains how to create and implement custom certificates.

Using Custom Security Certificates

You can use OpenSSL, Microsoft Certification Authority, or a public certificate authority (CA) of your choice to create your certificates. If you are not using OpenSSL, consult your certificate authority's documentation for instructions on creating certificates in a Windows Certificate Store-compatible format.

The procedures is this section use OpenSSL to generate certificates that will satisfy most security scanner tools when the root signing certificate is known to them.

> **Caution: Certificates are stored in the Windows Certificate Store**
>
> Certificates are stored in the Windows certificate store. If you have old certificates that are stored on the host, they should be deleted to avoid conflicts or confusion.

Custom Certificate Guidelines

If you choose to use your own certificates, follow these general guidelines:

- Save your root CA signing certificate in a safe place for deployment to clients.
- Back up private and public keys to secure locations.
- Never store files created when generating keys or certificates on network drives without password protection.

- Once certificates have been deployed to the Windows certificate store, the files they came from are no longer needed and can be deleted.
- Standard automatic tools, such as Automatic Certificate Enrollment and Group Policy, can be used for deploying automatically generated certificates. Both Automatic Certificate Enrollment and Group Policies are implemented through Active Directory. See MSDN Active Directory documentation for more information.

Pre-session Encryption Algorithms

Connections are negotiated using the following supported RSA cipher suites:

- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_GCM_SHA384
- TLS_RSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA

Note: Minimum SSL version

These Max Compatibility security level cipher suites have a minimum required SSL version of TLS 1.2.

Contacting Support

If you encounter any problems installing, configuring, or running the Graphics Agent for macOS, you can create a support ticket with Teradici.

Before creating a ticket, be prepared with the following:

- A detailed description of the problem
- Your agent version number (how do I find my version number?)
- A prepared support file

The Teradici Community Forum

The PCoIP Community Forum enables users to have conversations with other IT professionals to learn how they resolved issues, find answers to common questions, have peer group discussions on various topics, and access the Teradici PCoIP Technical Support Service team. Teradici staff are heavily involved in the forums.

To visit the Teradici community, go to https://communities.teradici.com.

Finding the Graphics Agent for macOS Version Number

To find the installed version number of the Graphics Agent for macOS, open a Terminal window and run the following command:

```
sudo grep "Software Build ID" /var/log/pcoip-agent/agent.log
```

The command output will contain your installed version.

Creating a Technical Support File

Teradici may request a support file from your system in order to troubleshoot and diagnose PCoIP issues. The support file is an archive containing PCoIP Graphics Agent for macOS logs and other diagnostic data that can help support diagnose your problem.

To create a support bundle, open a Terminal window and run the following command:

```
sudo pcoip-support-bundler
```

Performing Diagnostics

Each PCoIP component creates and updates a log file which records its activity as the system is used. Most troubleshooting within a PCoIP system begins by examining these log files and looking for error conditions or other indications that may explain why the system is not operating as expected.

Log files for the Graphics Agent for macOS and other Teradici PCoIP components are saved to specific directories.

> **Note: Bundling log files for support**
>
> When investigating issues with Teradici support, you may need to provide a support file which includes system log files. Instructions are provided here.

Locating Agent Log Files

Log files for the PCoIP agent are located in the following directories by default. If you changed your agent's location during installation, the log files will be in your custom location instead.

Component	Log file location
Agent	`/var/log/pcoip-agent/agent.log`
Server/User	`/var/log/pcoip-agent/server.<user>.log`

> **Note: Bundling log files for support**
>
> When investigating issues with Teradici support, you may need to provide a support file which includes system log files. Instructions are provided here.

Setting Log Verbosity

Each PCoIP component generates diagnostic log messages. The default log levels are recommended for use in a production deployment. When troubleshooting a particular problem, Teradici Support Services may recommend adjusting the PCoIP event log verbosity level to obtain more information from certain parts of the system.

> **Note: This is a global setting**
>
> The `pcoip.event_filter_mode` directive is a global setting, and affects the output levels of all PCoIP components.

To change the log verbosity level, set the `pcoip.event_filter_mode` directive in the **`pcoip-agent.conf`** file. See Configuring the PCoIP Agent for instructions.

Session Log IDs

At the start of each PCoIP session, a unique session ID is generated by the PCoIP Client and passed to all connected PCoIP components (including the Graphics Agent for macOS). Log messages generated by the agent are prefixed with this session ID, making it easy to identify. All log messages generated during a single session, by any PCoIP component, will be prefixed with the same session log ID in RFC-4122 format:

```
yyyy-mm-ddThh:mm:ss.ffffffZ xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx > …
```

For example:

```
2015-11-06T08:01:18.688879Z 4208fb66-e22a-11d1-a7d7-00a0c982c00d > …
```

Log messages that do not pertain to a specific session will show a string of zeroes in place of the session log ID number.

If a PCoIP component does not receive a session log ID from the PCoIP client, or receives an invalid value, it will generate a new session log ID and distribute it to the other components in the system.

Troubleshooting License Issues

Teradici includes a license validation utility that scans your local system and any connected physical or cloud-based license servers for active licenses, and informs you of when your license subscription expires. For more information, see FAQ - Licensing Cloud Access Software in our Knowledge Base.

To run the license validation tool, type:

```
pcoip-validate-license
```

For more detailed information on `pcoip-validate-license`, type:

```
man pcoip-validate-license
```

To list your licenses and their expiration status, type:

```
pcoip-list-licenses
```

For more detailed instructions on `pcoip-list-licenses`, type:

```
man pcoip-list-licenses
```

Tracking Usage Over Time

Teradici Local License Server users can use our open-source script, which displays the maximum Cloud Access Software license concurrent usage for a license server over time. For more information, refer to our Github page.

Teradici Cloud Licensing users can write a short script that runs `pcoip-list-licenses` periodically (for example, every 60 minutes) on any PCoIP agent machine to track license usage.

Troubleshooting Connection Issues

If you are unable to connect to your remote machine using a PCoIP session, investigate the following possible causes.

Except where noted, the following checks apply to the remote macOS machine, and to the remote user account that will be accepting PCoIP connections.

Only one user can be logged in at a time

PCoIP connections will be denied if another user is already logged into the remote machine (for example, via the remote machine's user interface). This is true whether the logged-in user is using PCoIP or not. In order to ensure a remote user can log in, make sure all other users are logged out.

Make sure the remote user's name does not contain spaces

If your remote username contains spaces, the PCoIP agent will not be able to connect to it. You can either change the name to eliminate the space, or create a new account to use for PCoIP connections.

Confirm privacy permissions are correct

On the remote machine, log in using the account that will host PCoIP connections and check that the following permissions are granted on the remote machine:

> **Start the PCoIP agent before proceeding**
>
> If you have not already run the PCoIP agent, double-click on `/Applications/PCoIP Agent.app` to start it.

1. Open **System Preferences...** > **Security & Privacy**, and select **Accessibility** from the list of options.
2. Under ***Allow the apps below to control your computer***, confirm that **PCoIP Agent** appears in the list with a checkmark.

If *PCoIP Agent* does not have a checkmark:

a. Unlock the list of apps by clicking the lock icon in the bottom-left corner and providing your administrative credentials.

b. Grant permission to the PCoIP Agent by clicking the checkbox beside **PCoIP Agent**.

3. Next, select **Screen Recording** from the list of options, and confirm that the PCoIP Agent appears in the list and has a checkmark beside it.

   If *PCoIP Agent* does not have a checkmark:

   a. Unlock the list of apps by clicking the lock icon in the bottom-left corner and providing your administrative credentials.

   b. Grant permission to the PCoIP Agent by clicking the checkbox beside **PCoIP Agent**.

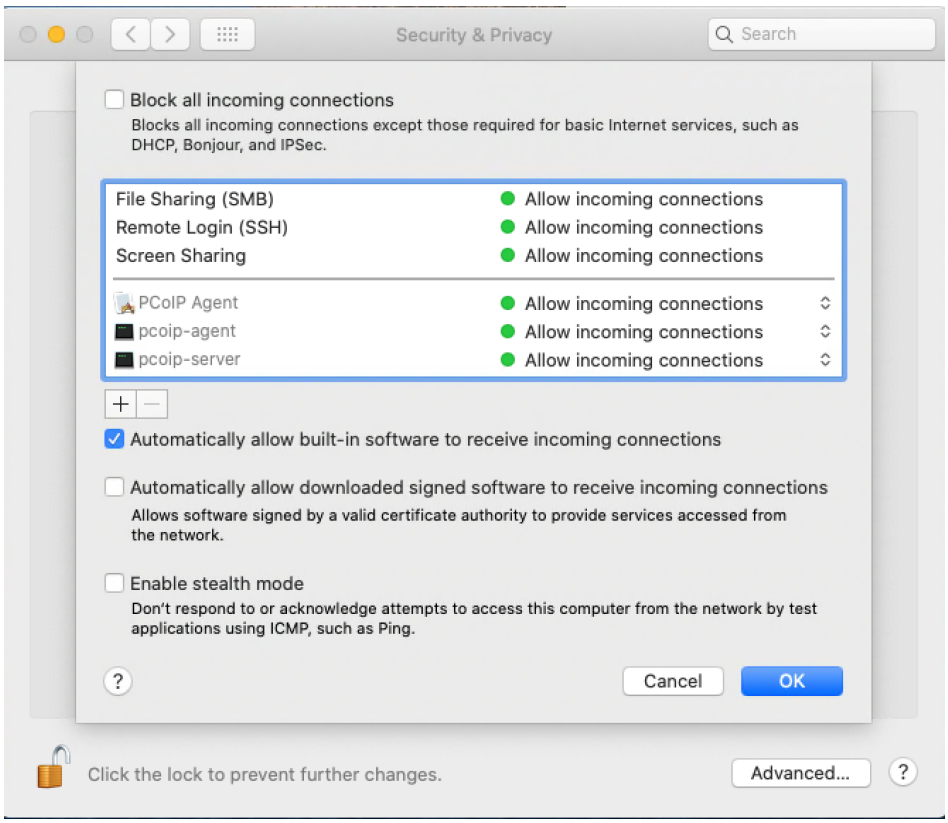
Confirm that a restart was performed

The remote machine must be restarted after the PCoIP agent is installed. Until it is restarted, PCoIP connections will fail.

Confirm network conditions

The remote and client machines must be able to reach one another. For both machines, one of these conditions must be met:

- The firewall is *off*, or
- The firewall is *on*, and:
  - **Automatically allow downloaded signed software to receive incoming connections** is checked, *or*
  - On the remote machine, **Automatically allow downloaded signed software to receive incoming connections** is *not* checked, *and* the following executables are allowed through the firewall:
    - `/Applications/PCoIP Agent.app`
    - `/Applications/PCoIP Agent.app/Contents/MacOS/pcoip-agent`
    - `/Applications/PCoIP Agent.app/Contents/MacOS/pcoip-server`



The *Automatically allow downloaded signed software to receive incoming connections* setting is in **System Preferences** > **Security & Privacy**, under the **Firewall** tab. Click **Firewall Options** to reveal it. The *Firewall Options* button is only visible if the firewall is on.

Check macOS login dialogs

Login prompts on the remote system will prevent a PCoIP session from starting. Make sure you have logged in to the remote computer using the user account that will host PCoIP sessions, and dismissed all prompts.

1. On the remote machine, log in to the user account that will host PCoIP sessions with either VNC or Screen Sharing.
2. From this connection, dismiss any login dialogs that are presented.
3. Attempt the PCoIP connection again, using the PCoIP Client.

Check for connected displays or display dongles

In some situations, connected displays or display dongles on the remote machine can prevent a PCoIP connection. This occurs because the real or simulated physical display interferes with the PCoIP agent's ability to create virtual displays for remoting.

Disconnecting removable displays and any display dongles from the remote machine may resolve this problem and allow a remote connection to be established.

Restart the PCoIP client

Quitting and re-launching the PCoIP client application may resolve connection issues.

Restart the remote machine

Rebooting the remote macOS system may resolve connection issues.

Reinstall the PCoIP Agent

In some situations, including after macOS updates, you may need to re-install the Graphics Agent for macOS to resolve connection issues.