# PCoIP Standard Agent for Linux Administrators' Guide

## 24.03

# Table of Contents

# Anyware Standard Agent for Linux 24.03

This guide is intended for administrators who are deploying the Standard Agent for Linux as part of HP Anyware. It assumes thorough knowledge of Linux conventions and networking concepts, including firewall configuration.

# About the Standard Agent for Linux

The Standard Agent for Linux is part of HP Anyware. It enables users to deliver virtual Linux desktops or custom applications to remote users. End users connect to their virtual desktops with a Anyware client, either directly or via a connection broker.

Typical end users of the Anyware Standard Agent for Linux include task workers and knowledge workers who need a Linux desktop, but do not require high-end GPU-powered graphics applications.

A deployed Standard Agent for Linux requires these components:

- **A host machine** which provides the desktop to remote clients. The host must be a virtual machine in a data center or in the cloud. See System Requirements for more information.
- **The agent software** installed on the host machine.

# Where to Find Information about Other Components

This guide describes the Standard Agent for Linux.

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

HP Anyware architectures and descriptions:

- PCoIP All Access Architecture Guide

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

- Anyware client 24.03 for Windows Administrators' Guide
- Anyware client 24.03 for macOS Administrators' Guide
- Anyware client for 24.03 Linux Administrators' Guide
- Tera2 Anyware Zero Client 24.03 Administrators' Guide

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

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

# What's New in This Release

**Release 24.03 of the Standard Agent for Linux includes:**

⚠ **RHEL/CentOS 7 is no longer supported**

CentOS 7 and RHEL 7 will reach end of life in June 2024. Support for these operating systems has been removed in HP Anyware 2024.03.

# System Requirements

The Standard Agent for Linux depends on the following system capacities and capabilities:

## Supported Instance Types

| VMware ESXi (6.0+) | KVM | AWS EC2 | Microsoft Azure | Google Cloud Platform |
|---|---|---|---|---|
| VMware Hardware Version 11 | QEMU/ KVM | Any instance type *that meets the instance requirements* | Any instance type *that meets the instance requirements* | Any instance type *that meets the instance requirements* |

# Host Instance Requirements

**⚠ RHEL/CentOS 7 is no longer supported**

CentOS 7 and RHEL 7 will reach end of life in June 2024. Support for these operating systems has been removed in HP Anyware 2024.03.

| Global instance requirements | |
|---|---|
| **Operating Systems** | • Ubuntu 22.04 LTS<br>• RHEL/Rocky Linux 8 |
| **Remote Host Memory** | At least *2GB* of RAM is required on the host desktop.<br>The agent should have at least *512MB* of available memory. |
| **Remote Host CPUs** | At least 2 CPUs are required on the host desktop.<br>Processors must support Streaming SIMD Extensions (SSE) 4.2.<br>*To use PCoIP Ultra, processors must support the AVX2 instruction set.* |
| **Network Ports** | The following ports must be open on the host desktop:<br>• TCP 443<br>• TCP 4172<br>• UDP 4172<br>• TCP 60443<br>Collaboration sessions require an open UDP port (default 64172) |
| **Storage** | At least 100MB for installation and 100MB for logging are recommended. |
| **User** | Cannot be `root`. You must create a user account for PCoIP connections. |

**⚠ Warning: Ubuntu 18.04 is obsolete**

Ubuntu 18.04 reached end of life on May 31, 2023, and is now obsolete. It is no longer a supported operating system for the Standard Agent for Linux.

**Using a standalone physical PC**

You can enable PCoIP connections to a standalone computer, without a discreet GPU, via the Standard Agent for Linux. Standalone physical PCs are currently not tested, but are expected to work. For more information and instructions, see HP Anyware Instructions for Standalone Computers in the HP Knowledge Base.

**Note: Elastic GPU and other EC2 instances supported**

The Standard Agent for Linux supports a variety of EC2 instances, including elastic GPU types such as eg1.large. Refer to Amazon EC2 Elastic GPUs documentation for more information.

# Feature Support

## Audio

The Standard Agent for Linux supports audio input and output between the host and the client. Audio can be enabled or disabled and audio bandwidth can be throttled by configuring the agent.

# Collaboration

**PCoIP Ultra Collaboration** enables a PCoIP session user to share their session with multiple remote collaborators using Anyware Software Clients.

> **Note: Collaboration terminology**
>
> When discussing this feature, we'll refer to the first user as the *session owner*, and subsequent users who join the session as *collaborators*. The session owner's screens, audio, and input devices (if allowed) are shared with the collaborators when they join the session.

Up to 5 collaborators can join an ongoing PCoIP session using the same invitation. The maximum number of collaborators can be reduced by configuring the agent.

> **Warning: Consider system resources**
>
> As the number of collaborators increases, the load on the Anyware agent's CPU, memory, and other resources will also increase. Test your system to ensure it can support all of your planned collaborators.

While connected, all collaborators can view and hear the session owner's screens and audio, and see the controlling collaborator's mouse movements. If permitted by the session owner, they can also share control of the session owner's keyboard and mouse using input control.

During a collaboration session, *all* of the session owner's desktop screens may be shared depending on the session owner's Anyware software client display settings. See Understanding Display Behavior for more information.

## Collaboration Requirements

PCoIP Ultra Collaboration is supported by all Anyware agents. Anyware software clients that support PCoIP Ultra can participate in collaboration sessions (all Anyware software clients 23.04 and higher meet this requirement).

Some collaboration features have specific version requirements for Anyware agents and Anyware software clients; these are noted below.

**Note: Only Anyware Software Clients are supported**

Anyware Tera2 Zero Clients and mobile clients do not support PCoIP Ultra and cannot join collaboration sessions.

## Feature Version Requirements

PCoIP Ultra Collaboration features depend on coordinated updates in Anyware agents and Anyware clients, so review these requirements carefully to ensure the features you need are supported by your system. **We strongly recommend using the latest versions of both Anyware agent and Anyware client**.

| Feature | Required versions | Notes |
|---|---|---|
| **Collaborate menu** | Anyware agent 23.06+<br>Anyware client 23.06+ | Both client and agent must be 23.06 or higher, with Collaboration and PCoIP Ultra enabled. |
| **Multiple collaborators** | Anyware agent 23.04+<br>Anyware client 23.04+ | HP Anyware versions 22.07–23.01 supported single collaborators only. |
| **Input control** | Anyware agent 23.01+<br>Anyware client 23.01+ | See Input Control for more information. |
| **Mouse visibility** | Anyware agent 22.07+<br>Anyware client 22.07+ | Session owner and collaborator software clients must be in *standard client mode* for mouse visibility to work. |

## Network Requirements

Each collaborator connection requires a separate UDP port. These ports are assigned in a range that *begins* with the configured UDP port (by default, 64172), and increments with each additional collaborator. All ports in this range must be open, both at the cloud provider network level and the local firewall at the host.

For example, using the default configuration and hosting three collaborators, the system would require inbound UDP ports 64172, 64173, and 64174 to be open.

| Maximum number of collaborators | Required inbound UDP ports |
|---|---|
| 1 | 64172 |
| 2 | 64172-64173 |
| 3 | 64172-64174 |
| 4 | 64172-64175 |
| 5 (default) | 64172-64176 |

You can change the starting port number if desired. If you change the configured starting UDP port, adjust these ranges and ensure your host firewall configuration allows traffic on the new ports.

For direct connections, or brokered connections that do not use a Security Gateway, each collaborator's Anyware client must be able to reach these ports. For brokered connections using a Security Gateway, only the Security Gateway must be able to reach them.

## Enabling Collaboration

**Important: Anyware agent steps**

Collaboration sessions are enabled and configured on the Anyware agent machine before starting collaboration sessions. Make sure the Anyware agent version you are using supports the collaboration features you expect. For details, see Feature Version Requirements.

The following steps apply to the session owner's desktop machine.

PCoIP Ultra Collaboration is enabled, disabled, and configured on the Anyware agent machine. It is disabled by default, and must be enabled by activating both *PCoIP Ultra* and *Collaboration* on the remote desktop.

**To enable PCoIP Ultra Collaboration:**

1. Open `/etc/pcoip-agent/pcoip-agent.conf` in a text editor.

2. If PCoIP Ultra is not already enabled, enable it by adding a new line in the text editor, and configuring the PCoIP Ultra offload mode:

| PCoIP Ultra mode | Supported by | Add a line with this value |
|---|---|---|
| **CPU Offload** | Standard Agents, Graphics Agents | `pcoip.ultra = 1` |
| **GPU Offload** | Graphics Agents | `pcoip.ultra = 2` |
| **Auto Offload** | Graphics Agents | `pcoip.ultra = 3` |

3. Add a new line enabling collaboration:

```
pcoip.enable_collaboration = 1
```

4. **Optional**: To enable *Collaboration Input Control* if desired, stay in the text editor and follow the instructions in Enabling Input Control below.
5. **Optional**: You can change the UDP starting port if needed (the default starting port is UDP 64172). To change the collaboration port number, add a new line specifying the new value:

```
pcoip.collaboration_udpport = <new_collaborator_port>
```

...where `<new_collaborator_port>` is your new starting port number.

6. Save the file and exit the editor.
7. Restart the Anyware Agent service:

```
sudo systemctl restart pcoip
```

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

## Input Control

*Collaborator input control* allows collaborators to use their own mice and keyboards to control the session owner's desktop. **This feature is disabled by default**, and must be enabled on the Anyware agent before it is available.

### ENABLING INPUT CONTROL

Input control is disabled by default, and the option to give collaborators input control cannot be selected.

To use Input Control, enable it on the Anyware agent. This change takes effect on the next PCoIP session:

1. Using a text editor, open the following file:

   ```
   /etc/pcoip-agent/pcoip-agent.conf
   ```

2. Add a new line enabling input control:

   ```
   pcoip.enable_collaboration_input_control = 1
   ```

3. **Optional:** Provide a custom input control timeout value (specified in milliseconds; 3000ms is 3 seconds):

   ```
   pcoip.collaboration_input_control_timeout <timeout_value_in_ms>
   ```

4. Save the file and exit the text editor.

#### DISABLING INPUT CONTROL

If Input Control has been enabled and you wish to disable it again:

1. Using a text editor, open the following file:

   ```
   /etc/pcoip-agent/pcoip-agent.conf
   ```

2. Disable input control:

   ```
   pcoip.enable_collaboration_input_control = 0
   ```

3. Save the file and exit the text editor.

# Configuring Collaboration

## Changing the Collaboration Starting Port

The default *starting* UDP Port for collaborator sessions is 64172. You can change this value if needed. Remember that you must also open a range of UDP ports that *begin* with this value to accommodate all of your collaborators; see Network Requirements for examples.

**To change the Collaboration session port:**

1. Open `/etc/pcoip-agent/pcoip-agent.conf` in a text editor.
2. Create new config entry specifying the new starting UDP port number to use:

   ```
   pcoip.collaboration_udpport = <new_collaborator_port>
   ```

3. Save the file and exit the text editor.
4. Restart the Anyware Agent service:

   ```
   sudo systemctl restart pcoip
   ```

## Changing the Maximum Number of Collaborators

PCoIP Ultra Collaboration supports up to 5 collaborators on the same PCoIP session. You can further limit the number of allowed collaborators by changing the *maximum collaborators* setting to a value from 1-5. By default, the system allows 5 collaborators.

**To Change the Maximum Number of Collaborators:**

1. Open `/etc/pcoip-agent/pcoip-agent.conf` in a text editor.
2. Create a new entry, replacing `<maximum number of collaborators>` with your desired value from 1-5:

   ```
   pcoip.max_collaborators = <maximum number of collaborators>
   ```

3. Save the file and exit the editor.
4. Restart the Anyware Agent service:

   ```
   sudo systemctl restart pcoip
   ```

### Changing the Input Control Timeout Value

Input control is released and made available to other collaborators by idling all input devices for a brief period. By default, this control timeout is 3 seconds (3000ms). You can change this value by configuring the Anyware agent machine as follows:

1. Using a text editor, open the following file:

   ```
   /etc/pcoip-agent/pcoip-agent.conf
   ```

2. Provide a custom input control timeout value (specified in milliseconds; 3000ms is 3 seconds):

   ```
   pcoip.collaboration_input_control_timeout = <timeout_value_in_ms>
   ```

3. Save the file and exit the text editor.
4. Restart the Anyware agent service:

   ```
   sudo systemctl restart pcoip
   ```

## Sharing Your Session With Collaborators

You can invite up to 5 collaborators to participate in your session, and optionally allow them to control your desktop.

> **Important: Anyware Client steps**
>
> Collaboration sessions are shared from Anyware clients in established PCoIP sessions. Make sure the software client version you are using supports the collaboration features you expect. For details, see Feature Version Requirements.

Collaboration sessions are managed using the **Collaboration manager**. The collaboration manager shows you who is connected to your session, whether each collaborators can view or control the session, and allows you to invite new collaborators or stop collaborating.

**Note: New Collaboration Manager menu option**

The Collaboration manager can now be launched by using the client's in-session menu, in addition to the system menu bar.

**To launch the Collaboration Manager:**

1. Connect to a PCoIP session with PCoIP Ultra and Collaboration enabled.
2. From the remote session, open the **Collaboration Manager** using either of these methods:

   - **From the in-session menu:** From the in-session menu, select **Collaborate** > **Invite to Collaborate**.

     Collaboration Manager Menu Option

     The Collaborate menu appears if both the Anyware agent and Anyware client are version 23.06 and higher, and if both support Collaboration. If the Collaborate menu does not appear, open the Collaboration Manager using the menu bar instead.

   - **From the menu bar:** by clicking the **Anyware Collaboration** icon in the menu bar:

     Launch collaboration manager

     This example shows a Linux desktop; yours may vary depending on which operating system you are connecting to.

To invite collaborators, the session owner generates a *collaboration invitation* using the collaboration manager, and distributes the invitation to all collaborators.

## About Collaboration Invitations

Collaboration invitations are created by the session owner and distributed to collaborators, who use them to join an established collaboration session.

**Note: About collaboration invitations**

A single collaboration invitation can be used by multiple collaborators (up to the maximum number configured). You do not need to generate a new invitation for each collaborator. Collaboration invitations behave as follows:

- If an invitation is generated but no collaborators connect within one hour, it expires and can no longer be used. If this happens, generate a new invitation.
- If *any* collaborators connect using an invitation, the invitation is activated and its time limit is removed. Once activated, an invitation can be re-used until the session owner stops collaborating or ends the session.
- Collaborators can disconnect from a collaboration session and then rejoin it later using the same invitation.
- Collaboration sessions persist even if all collaborators leave and only the session owner remains. Until the session owner disconnects or stops the collaboration session, collaborators can rejoin the session using the same invitation.
- The collaboration invitation remains valid until the session owner disconnects or stops the collaboration session.

## Inviting the First Collaborator

To begin a collaboration session, generate an invitation using the Collaboration Manager.

**To generate a collaboration invitation:**

1. Connect to a PCoIP session with PCoIP Ultra and Collaboration enabled.
2. From the remote session, open the **Collaboration Manager** using either of these methods:
   - **From the in-session menu:** From the in-session menu, select **Collaborate** > **Invite to Collaborate**.

     Collaboration Manager Menu Option

     The Collaborate menu appears if both the Anyware agent and Anyware client are version 23.06 and higher, and if both support Collaboration. If the Collaborate menu does not appear, open the Collaboration Manager using the menu bar instead.
   - **From the menu bar:** by clicking the **Anyware Collaboration** icon in the menu bar:

Launch collaboration manager

This example shows a Linux desktop; yours may vary depending on which operating system you are connecting to.

3. The Collaboration manager generates and displays an invitation:

   Collaboration invitation

   The invitation contains two pieces of information that are used to invite the collaborator:

   - **Invitation Link**: The collaborator will use this link to join your session. The link may be opened on any Mac, Windows or Linux machine with a Anyware Client 21.07 or newer.
   - **Invitation passcode**: 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.

4. Share the *invitation link* and the *invitation passcode* with the collaborator.

   - To share both the link and the code at once, click the **Copy invitation** button. This will create a single message containing both the link and the code and place it on your clipboard. Share this with your collaborators using any acceptable method.
   - To share the link and code *separately*, click the *copy* button beside each item and share them using separate communications. Sharing the invitation this way reduces risk in the event that a message is inadvertently sent, forwarded, or intercepted by a third party.

## Inviting Additional Collaborators

Once the collaboration session has been created, you can invite additional collaborators by sharing the same invitation link and passcode with them. You can also view the invitation, and copy its link and passcode for sharing, using the Collaboration Manager.

**To view and copy the invitation link and passcode:**

1. From the remote session, open the **Collaboration Manager** using either of these methods:

   - **From the in-session menu:** From the in-session menu, select **Collaborate** > **Invite to Collaborate**.

     Collaboration Manager Menu Option

The Collaborate menu appears if both the Anyware agent and Anyware client are version 23.06 and higher, and if both support Collaboration. If the Collaborate menu does not appear, open the Collaboration Manager using the menu bar instead.

- **From the menu bar:** by clicking the **Anyware Collaboration** icon in the menu bar:

  Launch collaboration manager

The collaboration manager shows a list of your active collaborators (if any).

2. In the Collaboration manager, below the list of active collaborators, click **Invite Collaborator**:

   Collaboration Manager with No Collaborators

3. The Collaboration manager displays the generated invitation. Note that this is the *same* invitation link and passcode you used previously. It is not a new invitation:

   Collaboration invitation

4. Share the *invitation link* and the *invitation passcode* with the additional collaborators.

## Accepting or Declining Collaborators

Once distributed, the session owner's collaborators can join the collaboration session. As collaborators use the invitation, the session owner is notified and can accept or reject each connection attempt.

**To respond to a collaborator**:

1. When the collaborator attempts to join the session, the Collaboration manager will display options to accept or reject the connection.

   Accept or reject the invitation

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

## Ending a Collaboration Session

The collaboration session will end when the session owner disconnects their PCoIP session, or if they stop collaborating using the collaboration manager.

Ending the collaboration session invalidates the invitation. To start a new session, generate a new invitation by inviting another collaborator.

**To stop collaborating:**

1. From the remote session, open the **Collaboration Manager** using either of these methods:
   - **From the in-session menu:** From the in-session menu, select **Collaborate** > **Invite to Collaborate**.

     Collaboration Manager Menu Option

     The Collaborate menu appears if both the Anyware agent and Anyware client are version 23.06 and higher, and if both support Collaboration. If the Collaborate menu does not appear, open the Collaboration Manager using the menu bar instead.
   - **From the menu bar:** by clicking the **Anyware Collaboration** icon in the menu bar:

     Launch collaboration manager

     This example shows a Linux desktop; yours may vary depending on which operating system you are connecting to.
2. Click the **Stop Collaboration** button.

   Stop collaborating

## Allowing Collaborators to Control the Session

*Collaborator input control* allows collaborators to use their own mice and keyboards to control the session owner's desktop. **This feature is disabled by default**, and must be enabled on the Anyware agent before it is available.

Once enabled, input control options are available from the collaboration manager. Input control can be granted (or retracted) for each user separately or for all users at once.

**Note: Disabling input control globally**

You can disable Input Control on the Anyware agent, which turns the feature off entirely. When disabled this way, session owners will not be able to allow collaborators to take control, and all sessions will be view-only. For more information, see Disabling Input Control.

> **Important: The session owner always has control of their Anyware client's in-session menu**
>
> The session owner always has control of their Anyware client's in-session menu. If the owner is unable to reclaim session input control for any reason, they can disconnect the PCoIP session using the in-session menu option. When the owner disconnects from the session, the collaborator is immediately disconnected.

### ENABLING INPUT CONTROL FOR COLLABORATORS

The following steps will allow one or more collaborators to take control of the session desktop. The collaborators will not immediately have control when this is granted; they must still take control using the process described above.

This option will not be available if Input Control has been disabled on the Anyware agent.

**To allow collaborators to control the session desktop**:

1. From the remote session, open the **Collaboration Manager** using either of these methods:

    - **From the in-session menu:** From the in-session menu, select **Collaborate** > **Invite to Collaborate**.

      Collaboration Manager Menu Option

      The Collaborate menu appears if both the Anyware agent and Anyware client are version 23.06 and higher, and if both support Collaboration. If the Collaborate menu does not appear, open the Collaboration Manager using the menu bar instead.

    - **From the menu bar:** by clicking the **Anyware Collaboration** icon in the menu bar:

      Launch collaboration manager

      This example shows a Linux desktop; yours may vary depending on which operating system you are connecting to.

2. Grant input control using one of these methods:

    - To allow input control for *all collaborators*, click the dropdown menu at the top right and select **All can control**.

      Grant control permission to all collaborators

    - To allow input for *one collaborator*, click the dropdown menu beside the collaborator's name and select **Can control**.

### STOPPING INPUT CONTROL

**To return a user (or all users) to view-only mode**:

1. From an active collaboration session, open the collaboration manager.
2. Beside the collaborator's name, click the dropdown menu and select **Can View**.

## Understanding Display Behavior

Collaboration sessions support sharing of multiple monitors, which varies by session owner's client setting as follows:

- **Windowed mode**: The session in the owner's client window will be shared.
- **Fullscreen One Monitor**: The single fullscreen session will be shared. The session owner should set their Anyware Client to *Fullscreen One Monitor* mode prior to starting the collaboration session.
- **Fullscreen All Monitors**: All monitors will be shared, beginning with the Session Owner's monitor 1 and continuing up to the number of displays in the collaborator's system. The monitors that are shared cannot be configured, and are shared in system order.

  *When using this mode, the session owner should assume that collaborators can see **all** displays unless a specific configuration has been tested and verified.*

  For example, if the session owner has four monitors and a collaborator only has two, the collaborator will see the session owner's first and second monitors. If different collaborators have a different number of screens, each will see as many displays their system supports; in this scenario, you may have some displays that are visible to certain collaborators but not others.

  The session owner should set their Anyware Client to *Fullscreen All Monitors* mode prior to starting the collaboration session.

If the session owner's and collaborator's screen resolutions are different, the collaborator's screen will use scrollbars and letterboxing to display the shared content.

If *high performance client* mode is enabled, and if the session owner's resolution is greater than the collaborator's, the collaborator's screen will be clipped instead.

# Joining a Collaboration Session

Collaborators receive the invitations generated by session owners, and use the invitation URI and passcode to connect to the session.

**Important: Anyware Client steps**

Collaborators join sessions using Anyware clients. Make sure the software client version you are using supports the collaboration features you require. For details, see Feature Version Requirements.

Each collaborator can join the session with the collaboration link and the Collaboration Invitation passcode. The same URI and passcode are used for all collaborators on the same session.

**To join a collaboration session as a collaborator:**

1. Open a web browser and go to the collaboration link 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 *Anyware Client* application. Allow the browser to open the Anyware client.
3. When the Anyware client opens, it will prompt you for your name and the Collaboration Invitation passcode. The name you provide here will identify you in the collaboration session. The collaboration invitation passcode is the six digit number provided by the session owner. Enter both values and click **Submit**.
4. Once the session owner accepts your connection request, the Collaboaration screen share will start.
5. To leave the collaboration session, select **Connection** > **Disconnect** from the Anyware Client menu.

## Collaborator Input Control

If the session owner has enabled input control for a collaborator, the collaborator can take control of the session owner's desktop including mouse, keyboard, and pointer activity. The session owner retains the ability to stop input control at any time.

### USING INPUT CONTROL AS A COLLABORATOR

A collaborator who has input control can release it by idling—stopping all keyboard, mouse, and pointer activity—for a short time. Once the control timeout has elapsed, the floor is open, and whichever collaborator provides input next takes control.

By default, the control timeout is 3 seconds. The timeout value can be configured when enabling the input control feature.

For example: the session owner has initial control of the session. In order to give control to the collaborator, the owner takes their hands off the keyboard and mouse for three seconds, allowing the control timeout to pass. A collaborator then moves their mouse, which gives them control. To give control back to the session owner, the collaborator takes their hands off their keyboard and mouse for three seconds. This exchange continues as long as needed.

# Supported Displays

The Standard Agent for Linux supports a maximum of four displays on the Anyware client, and a maximum resolution of 4K UHD (3840×2160).

Monitors can be arranged in a vertical line, a horizontal line, or as a 2×2 box display. They can be used in any standard rotation (0°, 90°, 180°, or 270°), with any monitor as the primary display.

**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.

**Important: Attaching monitors to the host machine in not supported**

Anyware client supports a maximum of four displays. Attaching extra monitors to the host machine will conflict with client display topologies.

# PCoIP Ultra

The Standard Agent for Linux provides support for PCoIP Ultra. 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 now defaults to "Auto Offload" on Graphics agent machines**, provided that both the client machines and the agent machines are capable of supporting CPU Offload as well as GPU Offload. Additionally, **YUV Chroma subsampling defaults to 4:2:0**. This ensures higher framerates of graphics and optimized motion content, while ensuring efficient utilization of bandwidth.

**Note: PCoIP Ultra Default Value**

*For most users, the default PCoIP Ultra value will provide the best possible experience.* Carefully review the recommended use cases in the next section to determine whether you should change the PCoIP Ultra value.

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 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.

For *all other scenarios*, we recommend that you leave PCoIP Ultra disabled.

# Requirements

To take advantage of PCoIP Ultra, you need:

- An **Anyware agent** (any supported version)
- An **Anyware Software Client** (any supported version)

> **Anyware Tera2 Zero Clients are not supported**
>
> PCoIP Ultra is supported by Anyware Software Clients only. Anyware Tera2 Zero Clients cannot use PCoIP Ultra.

- The CPUs on both the agent and the client machines must support the AVX2 instruction set.

# Enabling PCoIP Ultra

To enable PCoIP Ultra features, edit the `pcoip-agent.conf` file and set the `pcoip.ultra` configuration setting as required:

- **1** To turn on *PCoIP Ultra CPU Offload*. CPU offload requires CPU support for the AVX2 instruction set on both the remote host and client. The Anyware 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.

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

> **Turning PCoIP Ultra off**
>
> To disable PCoIP Ultra, set `pcoip.ultra` to 0.

> **Setting configuration values**
>
> If you don't know how to set Anyware agent configuration values, refer to Configuring the Standard Agent for Linux.

# Printing

The Standard Agent for Linux does not support printing from local printers connected to Anyware clients. Similarly, printing from USB printers that are connected by means of the USB Bridging feature is also not supported.

The following printing options are available:

- Linux agents can print to any printer on the host agents' local area network.
- If your agent 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

## USB Overview

The Standard Agent for Linux provides support for USB devices and certain Wacom tablets attached to Anyware clients.

**USB bridging must be explicitly installed**

When installing the Standard Agent for Linux, you must explicitly enable support for USB bridging by installing the required USB dependencies yourself. Refer to the installation steps for Ubuntu and RHEL or Rocky Linux for the required commands. If the required USB dependencies are not installed, the Standard Agent for Linux will be incapable of bridging USB devices.

This requirement does not affect support for keyboards, and mice or other pointer devices. It does not affect Wacom tablet support.

If the required USB packages are installed, USB bridging support is enabled by default. Administrators can disable or configure USB behavior by changing configuration options.

Keyboards, mice, and other pointer devices are managed by Anyware clients, and are always allowed.

### Xbox One Controller Support

The Standard Agent for Linux supports Xbox One controllers when attached to Anyware Zero Clients.

**Supported by Anyware Zero Clients only**

This feature is supported only by Anyware Zero Clients. It is not currently supported by Anyware Software Clients.

The following Xbox One controllers are supported:

- Xbox One 2015
- Xbox One
- Xbox One S
- Xbox One Bt
- Xbox One Elite

# Tangent Panel Support

The following Tangent panels are supported when connecting from a Windows or Linux software client to a Windows or Linux agent (both 23.04 or higher).

- Tangent Ripple
- Tangent Wave
- Tangent Element BT
- Tangent Element MF
- Tangent Element KB
- Tangent Element TK
- Tangent Arc (Navigation)
- Tangent Arc (Grading)
- Tangent Arc (Function)

The Graphics Agent for macOS and the Software Client for macOS do not support Tangent panels.

# Wacom Tablets

The Standard Agent for Linux supports Wacom tablets in two configurations: *bridged*, where peripheral data is sent to the desktop for processing, and *locally terminated*, where peripheral data is processed locally at the Anyware client.

## 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 Anyware agent, Anyware software client, and Anyware Zero Client firmware. To find out when support was added for individual Wacom device, refer to the release notes for your client.

> ⚠ **Caution: Using Wacom Local Termination on Ubuntu Cloud Hosts**
>
> Cloud-based Ubuntu hosts may fail to properly handle locally terminated Wacom tablets. When this occurs, pressure sensitivity and other advanced features will not work properly. To correct this issue, follow this procedure.

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

### Anyware client support for *locally terminated* Wacom tablets and the Standard Agent for Linux

| | Anyware Tera2 Zero Client | Anyware Software Client for Windows | Anyware Software Client for macOS | Anyware Software Client for Linux |
|---|---|---|---|---|
| **Intuos Pro Small**<br>*PTH-460* | — | ✔ | ✔ | ✔ |
| **Intuos Pro Medium**<br>*PTH-660* | ✔[1] | ✔ | ✔ | ✔ |
| **Intuos Pro Large**<br>*PTH-860* | ✔[1] | ✔ | ✔ | ✔ |
| **Cintiq Pro 16**<br>*DTH-167* | — | ✔ | ✔ | ✔ |
| **Cintiq Pro 16**<br>*DTH-1621* | — | ✔ | ✔ | ✔ |
| **Cintiq 22**<br>*DTK-2260* | — | ✔ | ✔ | ✔ |
| **Cintiq 22HD**<br>*DTK-2200* | ✔[2] | ✔ | — | ✔ |
| **Cintiq 22HDT - Pen & Touch**<br>*DTH-2200* | ✔[2] | — | — | — |
| **Cintiq Pro 24**<br>*DTK-2420* | ✔[2] | ✔ | — | ✔ |
| **Cintiq Pro 24 - Pen & Touch**<br>*DTH-2420* | ✔[2] | ✔ | ✔ | ✔ |
| **Cintiq Pro 27**<br>*DTH-271* | — | ✔ | ✔ | ✔ |
| **Cintiq 32 Pro - Pen & Touch**<br>*DTH-3220* | ✔[3] | ✔ | ✔ | ✔ |

**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.

## Bridged Wacom Tablets

Bridged Wacom tablets are supported only in low-latency environments. Tablets in network environments with greater than 25ms latency will show reduced responsiveness and are not recommended.

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

### Anyware client support for *bridged* Wacom tablets and the Standard Agent for Linux

| | Anyware Tera2 Zero Client | Anyware Software Client for Windows | Anyware Software Client for macOS | Anyware Software Client for Linux |
|---|---|---|---|---|
| **Intuos Pro Small**<br>*PTH-460* | ✔ | ✔ | ✔ | ✔ |
| **Intuos Pro Medium**<br>*PTH-660* | ✔ | ✔ | ✔ | ✔ |
| **Intuos Pro Large**<br>*PTH-860* | ✔ | ✔ | ✔ | ✔ |
| **Cintiq Pro 16**<br>*DTH-167* | — | ✔ | ✔ | ✔ |
| **Cintiq Pro 16**<br>*DTH-1621* | — | ✔ | ✔ | ✔ |
| **Cintiq 22**<br>*DTK-2260* | ✔ | ✔ | ✔ | ✔ |
| **Cintiq 22HD**<br>*DTK-2200* | ✔ | ✔ | ✔ | ✔ |
| **Cintiq 22HDT - Pen & Touch**<br>*DTH-2200* | ✔<br>*Ubuntu only* | ✔<br>*Ubuntu only* | ✔<br>*Ubuntu only* | ✔<br>*Ubuntu only* |
| **Cintiq Pro 24**<br>*DTK-2420* | ✔ | ✔ | ✔ | ✔ |
| **Cintiq Pro 24 - Pen & Touch**<br>*DTH-2420* | ✔ | ✔ | ✔ | ✔ |
| **Cintiq Pro 27**<br>*DTH-271* | — | ✔ | ✔ | ✔ |
| **Cintiq 32 Pro - Pen & Touch**<br>*DTH-3220* | ✔ | ✔ | ✔ | ✔ |

Other Wacom tablets may work, but have not been tested.

**⚠ Caution: Do Not Calibrate Pen Displays from Wacom Center**

We recommend that you **NOT** calibrate your pen display from Wacom Center on the host machine. Doing so might result in the cursor getting offset when the tablet is used during a PCoIP session.

1. Local termination for Intuos Pro Small and Intuos Pro Medium requires Tera2 Zero Client firmware 6.2.0 or higher. ↩↩
2. Local termination for Cintiq 22HD, 22HDT, 24P, and 24PT requires Tera2 Zero Client firmware 6.5.0 or higher. ↩↩↩↩
3. Local termination for Cintiq Pro 32PT requires Tera2 Zero Client firmware 20.04 or higher. ↩
4. Launching a PCoIP session with a *bridged* Cintiq 22HDT (DTH-2200) and a *RHEL or Rocky Linux host* can cause the remote system to disconnect and become unresponsive. This issue does not occur when bridging to Ubuntu hosts. ↩

# Installation Guide

## Installing on Ubuntu

### Installing the Anyware Standard Agent for Linux on Ubuntu

Before you proceed with installation, a few prerequisites must be met.

#### Prerequisites

These instructions assume you have already built the remote desktop machine, and that the machine meets the agent's requirements.

**Important: A desktop environment is required**

Before proceeding, install a desktop environment of your choice. Kubuntu distributions are bundled with KDE; you can install KDE from other distributions by using this command:

```
sudo apt install kubuntu-desktop
```

To install Mate Desktop, use this command:

```
sudo apt install ubuntu-mate-desktop
```

These commands are provided as a convenience; there is no requirement for KDE or Mate Desktop. Any desktop environment will work.

A few other things to confirm before proceeding:

- SSH must be enabled.
- You must have a license registration code for the agent instance from HP (as part of a HP Cloud Access subscription).

- The desktop machine requires the following ports to be open: TCP 443, TCP 60443, TCP 4172, and UDP 4172.
- You must have super user (root) privileges and be able to issue `sudo` commands.
- If you are using a Local License Server, Local License Server, you'll need to know it's URL and port numbers.

> **Important: Protect your license registration code**
>
> The license registration code you receive from HP is unique to your organization, and should be protected as you would any sensitive data.
>
> Be careful that you do not inadvertently expose your registration code in forums or other public areas by pasting log messages without redacting sensitive information.

## Installation Overview

Once your prerequisites are in place, you can proceed with installation. Here's a brief overview of the process:

1. Connect to the machine using SSH.
2. Install the Anyware Agent.
3. If required, configure the agent software.
4. Disconnect the SSH session.
5. Connect to the desktop using a Anyware client.

If you're ready to start, connect to your machine with an SSH client and proceed to Install the Standard Agent for Linux.

# Installing the Standard Agent for Linux on Ubuntu

**Important: Required ports will be automatically opened**

The Standard Agent for Linux installer will add firewall exceptions for the following required PCoIP ports during installation: TCP 443, TCP 4172, UDP 4172, and TCP 60443.

**Important: IP Configuration Considerations**

For the Anyware Agent installation to complete successfully, the RHEL/Rocky 8 host machine must be configured to use both IPv4 and IPv6.

1. Download and install the repository, via the shell script provided here.
2. **Optionally** install USB dependencies, if you intend to support USB devices other than keyboards, mice, and pointer devices. *If you skip this step, USB redirection will be completely disabled and bridged USB devices will not work.*

   ```
   sudo apt install usb-vhci-dkms
   ```

3. Install the Anyware Standard Agent for Linux:

   ```
   sudo apt update
   sudo apt install pcoip-agent-standard
   ```

4. Note your machine's local IP address. Clients connecting directly to the host workstation will need this number to connect.
5. Enter the license registration code you received from us.

   **Note: These instructions are for Cloud Licensing**

   These instructions assume you are using Anyware Cloud Licensing to activate your PCoIP session licenses. If you are using the License Server instead, see Licensing the Standard Agent for Linux.

For unproxied internet connections, type:

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

For proxied internet conections, type:

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

6. Reboot the desktop.

**Note: Desktop user interfaces will only be available using PCoIP**

Once installed and running, the Anyware Standard Agent for Linux takes over the graphics subsystem which is then unavailable to hypervisors. You can only view the graphical user interface when connecting with a Anyware client.

For example, you cannot view an ESXi virtual machine console through VSphere; you must connect to the machine using PCoIP.

## GNOME Display Manager Support

**GNOME Display Manager (GDM)** is now supported by the Standard Agent for Linux as a credential authenticator when gdm-runtime-config is available. This allows PCoIP sessions to be locked and unlocked within the remoted session. When the PCoIP service starts, GDM is configured to run without obstructing access to the GPU. This configuration is reverted when the PCoIP service stops. This behavior is always on and requires no configuration on the Anyware agent. For more information on GDM, see the GNOME Display Manager Reference Manual.

## Installing USB Drivers on Secure Boot Enabled Linux Machines

If Secure Boot is enabled on a Linux host machine deployed on an ESXi resource, USB drivers cannot be installed on the host machine. This prevents the client-side USB devices from getting bridged to the host machine. To prevent this from occurring, perform the following steps, provided that **you have access to the UEFI Firmware Menu**.

**Note: If You Do Not Have UEFI Firmware Menu Access**

If you do not have access to the UEFI Firmware Menu, either disable Secure Boot or deploy the Virtual Machine without Secure Boot.

1. Connect to the remote host machine via SSH.
2. Run the following command:

   ```
   sudo mokutil --import /var/lib/dkms/mok.pub
   ```

3. If the Machine Owner Key (MOK) is not set, the mokutil utility will prompt you to create one.
4. Reboot the machine.
5. Enroll the MOK in the UEFI firmware menu.
6. Reboot the machine again.
7. Run the following commands:

   ```
   lsmod | grep usb
   dmesg | grep vhci
   ```

The output of this command indicates that `usb-vhci` is installed on the host machine.

## After Installation

Once you've installed the software, you can configure it, register licenses, or connect to it.

# 2. License the Agent

The Standard Agent for Linux 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 Anyware client.

You receive a registration code when you purchase a pool of licenses from HP. 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`

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

If you need to purchase licenses, contact HP.

## Troubleshooting Licensing Issues

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

## Using HP Cloud Licensing

To use Cloud Licensing, all you need to do is provide a registration code for each Anyware 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:

```
pcoip-register-host --registration-code=<registration-code> [--proxy-
server=<proxy-server-address>] [--proxy-port=<proxy-port-number>]
```

**Important: Allowlist network blocks for Anyware Cloud Licensing**

If you are using Anyware Cloud Licensing, you will need to add the following to your allowlist:

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

If you use an IP-based allowlist, we recommend your IT team add the following network blocks to your allowlist:

- `IPv4: 185.146.155.64/27`
- `IPv6: 2620:122:f005::/56`

**Important: Migrating from the previous specification**

Previously, our allowlist specification looked like this:

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

If you have an existing implementation using an IP-based allowlist like this, we recommend you leave it in place until the new allowlist is active and tested.

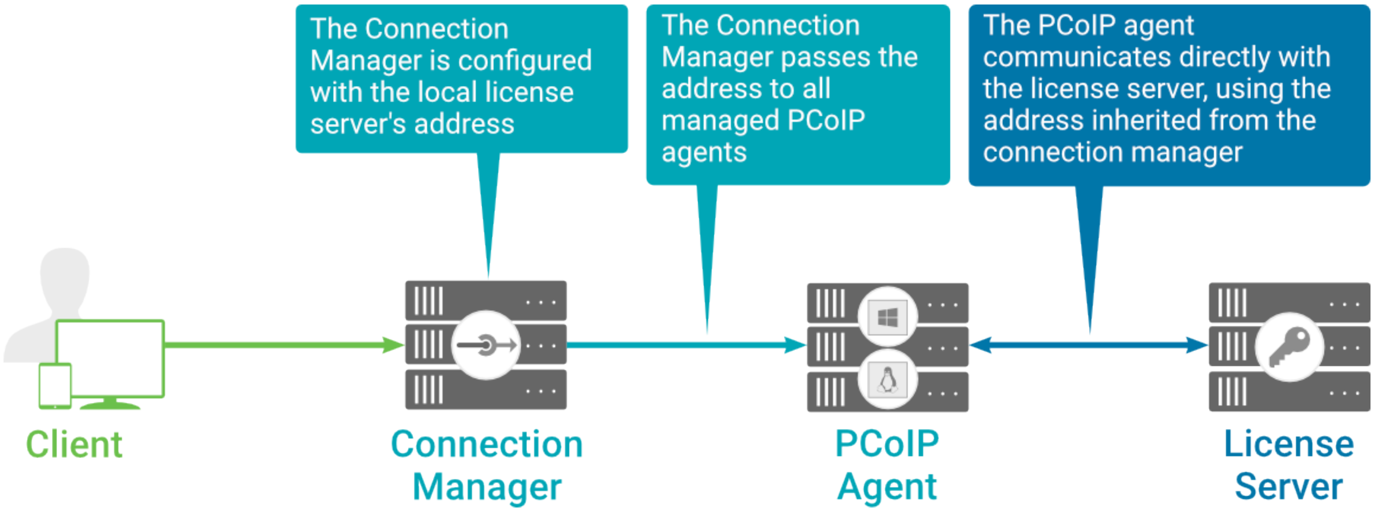
## Licensing Anyware Agents With a Local License Server

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

Configuring Anyware 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 Anyware clients connect directly to Anyware agents.

### BROKERED ENVIRONMENT LICENSING

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



The Connection Manager is configured with the local license server's address

The Connection Manager passes the address to all managed PCoIP agents

The PCoIP agent communicates directly with the license server, using the address inherited from the connection manager

**Local license validation using a Standard Agent for Linux and a brokered connection**

When using a Connection Manager, the license server address is only configured once no matter how many Anyware 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 Standard Agent for Linux machine. The command will ping the license server and attempt to retrieve information on an available license:

```
pcoip-validate-license ɆɆlicense-server-url <license-server-address> [ɆɆproxy-
server <proxy-server-address>] [ɆɆproxy-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 `Ɇ-proxy-server` and `Ɇ-proxy-port` parameters.

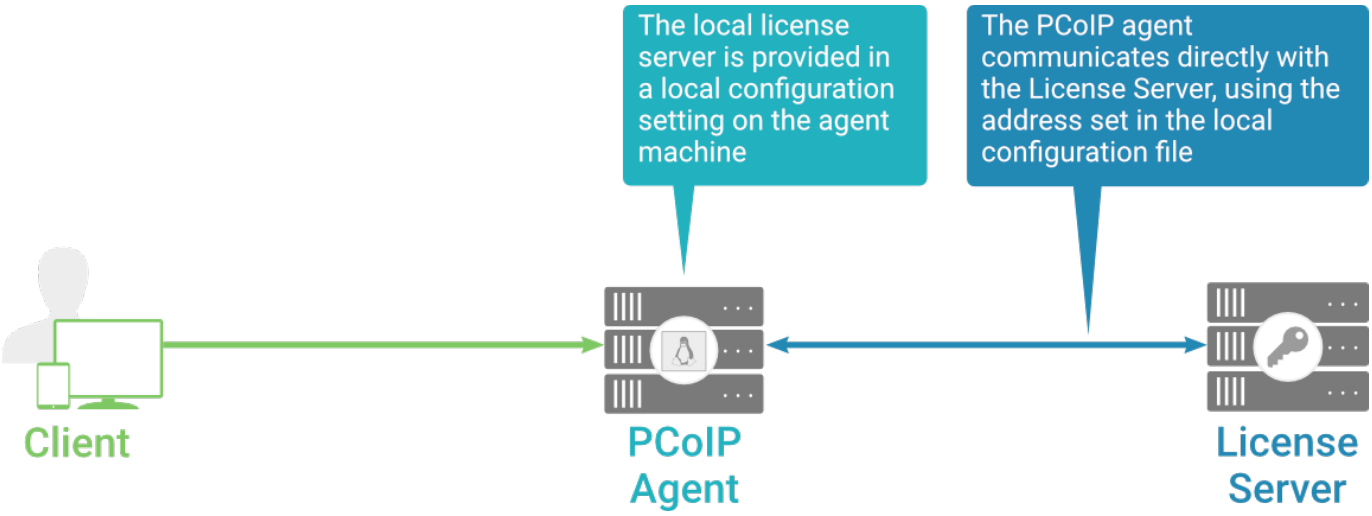
If successful, the response will show that a valid license was found on the license server, and its expiration date.

**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 Anyware agent is configured with the license server address via a local agent setting. When a client initiates a new PCoIP session, the Anyware agent uses its local configuration to communicate with the license server.


The local license server is provided in a local configuration setting on the agent machine
The PCoIP agent communicates directly with the License Server, using the address set in the local configuration file
Client
PCoIP Agent
License Server

**Local license validation using a Standard Agent for Linux and a direct (unbrokered) connection**

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

**To configure the License Server URL on the Standard Agent for Linux machine:**

1. Using a text editor, open `/etc/pcoip-agent/pcoip-agent.conf`.

2. Add or modify the `pcoip.license_server_path` directive:

   ```
   pcoip.license_server_path = <license-server-address>
   ```

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

3. If the license server is behind a proxy server, provide the proxy information using the `pcoip.license_proxy_server` and `pcoip.license_proxy_port` directives.
4. Save and close `pcoip-agent.conf`.

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 Standard Agent for Linux machine. The command will ping the license server and attempt to retrieve information on an available license:

```
pcoip-validate-license ɆɆlicense-server-url <license-server-address> [ɆɆproxy-
server <proxy-server-address>] [ɆɆproxy-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 `Ɇ-proxy-server` and `Ɇ-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 Standard Agent for Linux on Ubuntu

Updates to the Standard Agent for Linux will be published on a regular basis. New stable builds will be produced approximately every three months. Your configuration settings will be preserved after the upgrade.

Before you begin, make sure that you have read the Installation Overview topic.

1. Obtain the new Standard Agent installer.
2. Run the following three commands:

```
sudo apt update
sudo apt install pcoip-agent-standard
sudo reboot
```

# Uninstalling the Standard Agent for Linux

You can remove the Standard Agent for Linux from your system, or you can remove the repo config entirely.

## Remove the Standard Agent for Linux package

To remove the package, open a console window and run the following command:

```
sudo apt-get remove pcoip-agent-*
```

## Remove the repo configuration

If you want to remove the repo configuration completely, you can do that as well. You'll need to do this if you are switching from one channel to another (for example, from beta to stable), before reconfiguring with the new repo:

Replace `<repo_channel>` with one of the following, depending on the configuration you need to remove:

- `teradici-pcoip-agent.list`: Removes *stable* channel. Most users are on this channel, which contains our stable releases.
- `teradici-pcoip-agent-beta.list`: Removes *beta* channel.
- `teradici-pcoip-agent-dev.list`: Removes *dev* channel.

```
rm /etc/apt/sources.list.d/<repo_channel>.list
apt-get clean
rm -rf /var/lib/apt/lists/*
apt-get update
```

# Installing on RHEL or Rocky Linux

## Installing the Anyware Standard Agent for Linux on RHEL or CentOS

Before you proceed with installation, a few prerequisites must be met.

### Prerequisites

These instructions assume you have already built the remote desktop machine, and that the machine meets the agent's requirements.

Before proceeding with Standard Agent for Linux installation, install a desktop environment. To install a desktop environment in RHEL or CentOS, use the following command:

```
sudo yum groupinstall 'Server with GUI'
```

A few other things to confirm before proceeding:

- SSH must be enabled.
- You must have a license registration code for the agent instance from HP (as part of a HP Cloud Access subscription).
- The desktop machine requires the following ports to be open: TCP 443, TCP 60443, TCP 4172, and UDP 4172.
- You must have super user (root) privileges and be able to issue `sudo` commands.
- If you are using a Local License Server, Local License Server, you'll need to know it's URL and port numbers.

**Important: Protect your license registration code**

The license registration code you receive from HP is unique to your organization, and should be protected as you would any sensitive data.

Be careful that you do not inadvertently expose your registration code in forums or other public areas by pasting log messages without redacting sensitive information.

## Installation Overview

Once your prerequisites are in place, you can proceed with installation. Here's a brief overview of the process:

1. Connect to the machine using SSH.
2. Install the Anyware Agent.
3. If required, configure the agent software.
4. Disconnect the SSH session.
5. Connect to the desktop using a Anyware client.

If you're ready to start, connect to your machine with an SSH client and proceed to install the Standard Agent for Linux.

# Installing the Standard Agent for Linux on RHEL or CentOS

> **Important: Required ports will be automatically opened**
>
> The Standard Agent for Linux installer will add firewall exceptions for the following required PCoIP ports during installation: TCP 443, TCP 4172, UDP 4172, and TCP 60443.

> **Important: IP Configuration Considerations**
>
> For the Anyware Agent installation to complete successfully, the RHEL/Rocky 8 host machine must be configured to use both IPv4 and IPv6.

## Installing the Standard Agent for Linux in an Online Environment

Follow the steps in this section for installing the Standard Agent in an internet-connected environment.

1. Download and install the repository via the shell script provided here.
2. Install the EPEL repository:

   ```
   sudo yum install epel-release
   ```

3. If you want to install USB dependencies, the `usb-vhci kernel` modules need be built for which `kernel-devel` and `kernel-headers` that correspond to the version of the running kernel must be installed.

   To do this:

   a. Upgrade the system to get the latest kernel and install `kernel-devel` and `kernel-headers` while installing `usb-vhci`:

      ```
      sudo yum upgrade
      ```

      Run this command if you don't want to upgrade the kernel:

      ```
      yum install kernel-devel-$(uname r) kernel-headers-$(uname -r)
      ```

b. Reboot your machine to ensure the new kernel is running:

```
sudo reboot{}
```

c. Check the status of `kernel-devel` and `kernel-headers`:

```
rpm -qa kernel{, -devel, -headers} | sort
```

4. **Optionally** install USB dependencies, if you intend to support USB devices other than keyboards, mice, and pointer devices. **If you skip this step, USB redirection will be completely disabled and bridged USB devices will not work.**

```
sudo yum install usb-vhci
```

5. Install the Anyware Standard Agent for Linux:

```
sudo yum install pcoip-agent-standard
```

6. Note your machine's local IP address. Clients connecting directly to the host workstation will need this number to connect.
7. Enter the license registration code you received from us.

> **Note: These instructions are for Cloud Licensing**
>
> These instructions assume you are using Anyware Cloud Licensing to activate your PCoIP session licenses. If you are using the License Server instead, see Licensing the Standard Agent for Linux.

For unproxied internet connections, type:

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

For proxied internet conections, type:

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

8. Reboot the desktop.

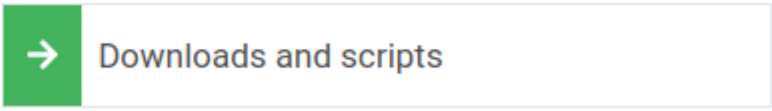
## GNOME Display Manager Support

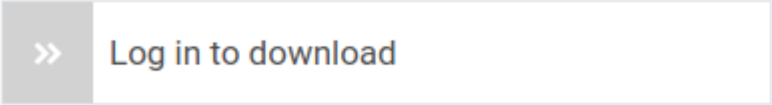
**GNOME Display Manager (GDM)** is now supported by the Standard Agent for Linux as a credential authenticator when gdm-runtime-config is available. This allows PCoIP sessions to be locked and unlocked within the remoted session. When the PCoIP service starts, GDM is configured to run without obstructing access to the GPU. This configuration is reverted when the PCoIP service stops. This behavior is always on and requires no configuration on the Anyware agent. For more information on GDM, see the GNOME Display Manager Reference Manual.

## Installing the Standard Agent for Linux in a Dark Site

The Standard Agent for Linux can be installed in dark site environments, also referred to as *offline environments*, that do not have a connection to the public internet. We provide archived bundles (tar.gz archives) for each supported operating system, which includes the core application and all dependencies.

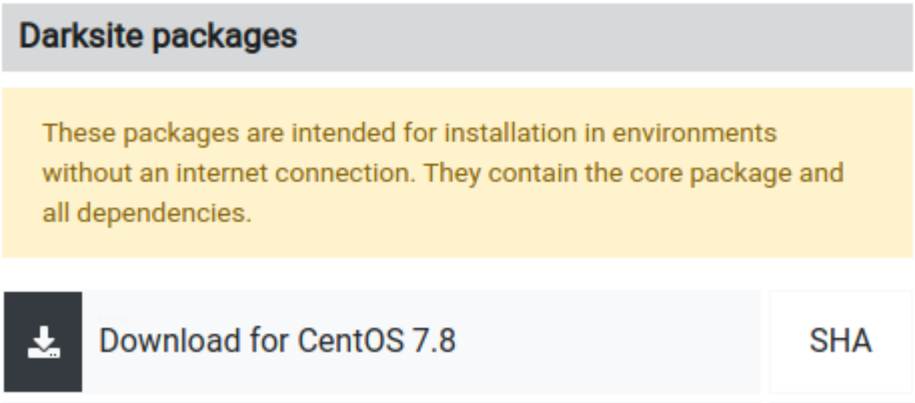
1. From an internet-connected machine, open a browser and navigate to HP's documents and downloads site. Look in the sidebar for documentation and download links.
2. Click the **Downloads and scripts** button.



> **Note: An account is required**
>
> If you are not logged in, you will see a log in prompt instead:
>
> 
>
>
> You can create an account when you click this button if you do not already have one.

3. Read and accept the End User License Agreement.
4. Under **Darksite packages**, find the download that matches your operating system, and click to download it. The installer is downloaded as a TAR file.

   *For brevity, this example shows only the Centos 7.8 package; all supported operating systems have an available download.*



Darksite packages

These packages are intended for installation in environments without an internet connection. They contain the core package and all dependencies.

5. Transfer the downloaded file to the production Linux machine using any acceptable method, such as a USB drive.
6. On the production machine, open a console window and navigate to the directory where you placed the installer.
7. If you want to install USB dependencies, the `usb-vhci kernel` modules need be built for which `kernel-devel` and `kernel-headers` that correspond to the version of the current Linux image must be installed on kernel image.

   To do this:

   a. Make sure that `kernel-devel` and `kernel-headers` corresponding to the current kernel are installed in your kernel image.

   b. Check the status of `kernel-devel` and `kernel-headers`:

   ```
   rpm -qa kernel{, -devel, -headers} | sort
   ```

8. Extract the installation files by running this command:

   ```
       sudo tar xvf <file path>/pcoip-agent-offline-rhel8.6_<pcoip
   version>.tar.gz
   ```

   where is the version of the Anyware Agent.

9. Install the agent by running the following command:

   - To install *with* USB device support:

   ```
   sudo ./install-pcoip-agent.sh pcoip-agent-standard usb-vhci
   ```

- To install *without* USB device support, omit the `usb-vhci` parameter:

  ```
  sudo ./install-pcoip-agent.sh pcoip-agent-standard
  ```

10. Provide the following when prompted:
    - **Agent type**: choose Standard Agent for Linux.
    - **USB device support**: If you will be allowing USB devices (other than keyboards, mice, and pointers), accept this option. If you install the agent without USB device support, only keyboards, mice, and pointers will work, and more sophisticated devices like Wacom tablets will act as pointing devices without any advanced functionality.
11. Update the kernel when prompted.
12. When prompted, reboot the machine.
13. Register your Standard Agent for Linux's license with your License Server. See Licensing Anyware Agents With a Local License Server for details.
14. Reboot the desktop.

## Installing USB Drivers on Secure Boot Linux Machines

If Secure Boot is enabled on a Linux host machine deployed on an ESXi resource, USB drivers cannot be installed on the host machine. This prevents the client-side USB devices from getting bridged to the host machine. To prevent this from occurring, perform the following steps, provided that **you have access to the UEFI Firmware Menu**.

**Note: If You Do Not Have UEFI Firmware Menu Access**

If you do not have access to the UEFI Firmware Menu, either disable Secure Boot or deploy the Virtual Machine without Secure Boot.

1. Connect to the remote host machine via SSH.
2. Run the following command:

   ```
   sudo mokutil --import /var/lib/dkms/mok.pub
   ```

3. If the Machine Owner Key (MOK) is not set, the mokutil utility will prompt you to create one.

4. Reboot the machine.
5. Enroll the MOK in the UEFI firmware menu.
6. Reboot the machine again.
7. Run the following commands:

```
lsmod | grep usb
dmesg | grep vhci
```

The output of this command indicates that `usb-vhci` is installed on the host machine.

## After Installation

Once you've installed the software, you can configure it, register licenses, or connect to it.

**Note: Desktop user interfaces will only be available using PCoIP**

Once installed and running, the Anyware Standard Agent for Linux takes over the graphics subsystem which is then unavailable to hypervisors. You can only view the graphical user interface when connecting with a Anyware client.

For example, you cannot view an ESXi virtual machine console through VSphere; you must connect to the machine using PCoIP.

# 2. License the Agent

The Standard Agent for Linux 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 Anyware client.

You receive a registration code when you purchase a pool of licenses from HP. 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`

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

If you need to purchase licenses, contact HP.

## Troubleshooting Licensing Issues

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

## Using HP Cloud Licensing

To use Cloud Licensing, all you need to do is provide a registration code for each Anyware 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:

```
pcoip-register-host --registration-code=<registration-code> [--proxy-
server=<proxy-server-address>] [--proxy-port=<proxy-port-number>]
```

**Important: Allowlist network blocks for Anyware Cloud Licensing**

If you are using Anyware Cloud Licensing, you will need to add the following to your allowlist:

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

If you use an IP-based allowlist, we recommend your IT team add the following network blocks to your allowlist:

- `IPv4: 185.146.155.64/27`
- `IPv6: 2620:122:f005::/56`

**Important: Migrating from the previous specification**

Previously, our allowlist specification looked like this:

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

If you have an existing implementation using an IP-based allowlist like this, we recommend you leave it in place until the new allowlist is active and tested.

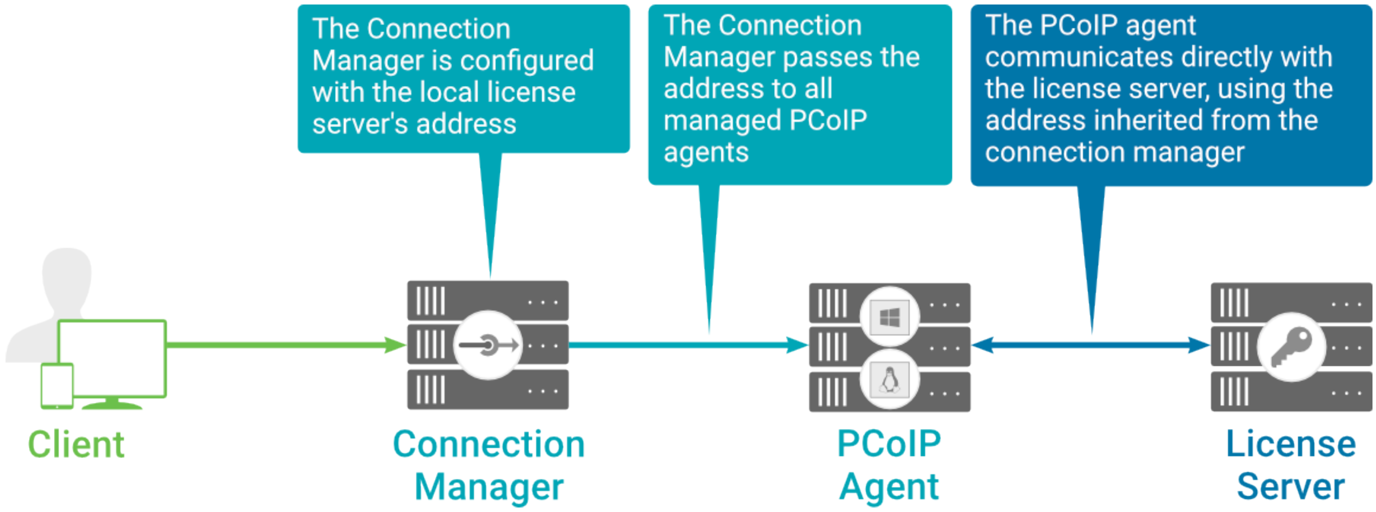
## Licensing Anyware Agents With a Local License Server

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

Configuring Anyware 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 Anyware clients connect directly to Anyware agents.

### BROKERED ENVIRONMENT LICENSING

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


The Connection Manager is configured with the local license server's address
The Connection Manager passes the address to all managed PCoIP agents
The PCoIP agent communicates directly with the license server, using the address inherited from the connection manager
Client
Connection Manager
PCoIP Agent
License Server

**Local license validation using a Standard Agent for Linux and a brokered connection**

When using a Connection Manager, the license server address is only configured once no matter how many Anyware 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 Standard Agent for Linux machine. The command will ping the license server and attempt to retrieve information on an available license:

```
pcoip-validate-license ɆɆlicense-server-url <license-server-address> [ɆɆproxy-
server <proxy-server-address>] [ɆɆproxy-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 `Ɇ-proxy-server` and `Ɇ-proxy-port` parameters.

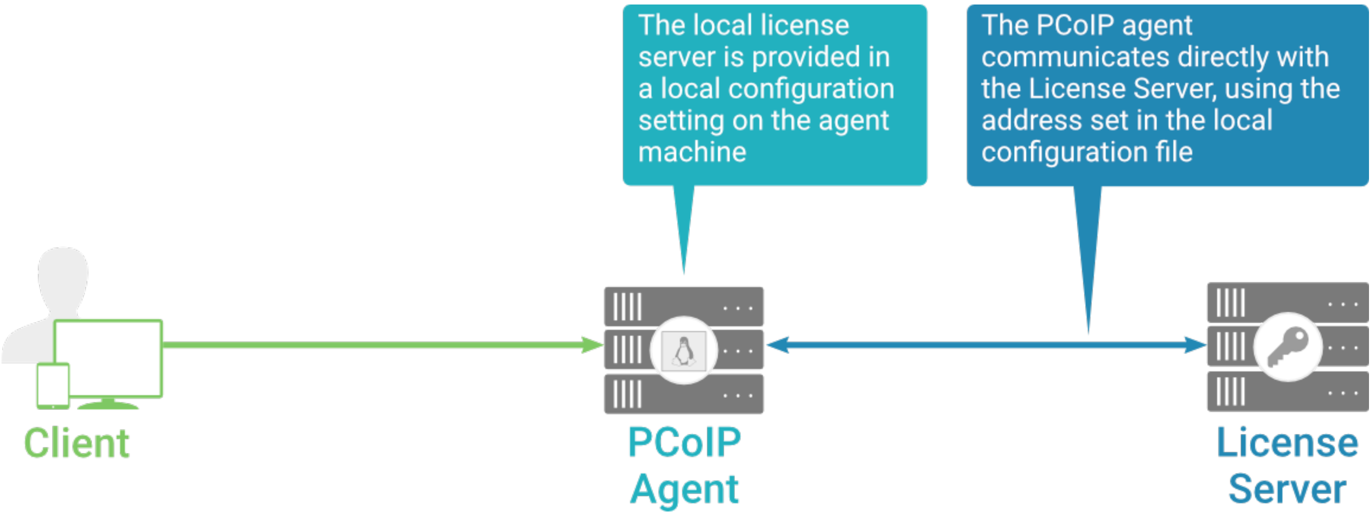
If successful, the response will show that a valid license was found on the license server, and its expiration date.

**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 Anyware agent is configured with the license server address via a local agent setting. When a client initiates a new PCoIP session, the Anyware agent uses its local configuration to communicate with the license server.


The local license server is provided in a local configuration setting on the agent machine
The PCoIP agent communicates directly with the License Server, using the address set in the local configuration file
Client
PCoIP Agent
License Server

**Local license validation using a Standard Agent for Linux and a direct (unbrokered) connection**

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

**To configure the License Server URL on the Standard Agent for Linux machine:**

1. Using a text editor, open `/etc/pcoip-agent/pcoip-agent.conf`.

2. Add or modify the `pcoip.license_server_path` directive:

   ```
   pcoip.license_server_path = <license-server-address>
   ```

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

3. If the license server is behind a proxy server, provide the proxy information using the `pcoip.license_proxy_server` and `pcoip.license_proxy_port` directives.
4. Save and close `pcoip-agent.conf`.

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 Standard Agent for Linux machine. The command will ping the license server and attempt to retrieve information on an available license:

```
pcoip-validate-license ɆɆlicense-server-url <license-server-address> [ɆɆproxy-
server <proxy-server-address>] [ɆɆproxy-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 `Ɇ-proxy-server` and `Ɇ-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 Standard Agent for Linux on RHEL or CentOS

Updates to the Standard Agent for Linux will be published on a regular basis. New stable builds will be produced approximately every three months.

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

Before you begin, make sure that you have read the Installation Overview topic.

To upgrade to the latest version, use the following three commands:

```
sudo yum makecache
sudo yum update pcoip-agent-standard
sudo reboot
```

# Uninstalling the Standard Agent for Linux

You can remove the Standard Agent for Linux from your system, or you can remove the repo config entirely.

## Remove the Standard Agent for Linux package

To remove the package, open a console window and run the following command:

```
sudo yum remove pcoip-agent-*
```

## Remove the repo configuration

If you want to remove the repo configuration completely, you can do that as well. You'll need to do this if you are switching from one channel to another (for example, from beta to stable), before reconfiguring with the new repo:

```
rm /etc/yum.repos.d/pcoip-agent.repo
rm /etc/yum.repos.d/pcoip-agent-source.repo
```

## Restore the gdm service

Uninstalling the Graphics Agent doesn't reinstate the previous system settings, which is why after the uninstallation completes, you must enable `gdm.service`.

Run the following command to restore the gdm service:

```
$ sudo systemctl enable gdm.service
```

# Configuration Guide

You can configure the Anyware agent, and optimize PCoIP protocol behavior for local network conditions, by adjusting configuration directives found in `/etc/pcoip-agent/pcoip-agent.conf`.

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

```
man pcoip-agent.conf
```

## Applying Configuration Changes

To set or change a configuration value, add or modify directives in `pcoip-agent.conf`. Place one directive on each line, in this format:

```
directive.name = <value>
```

For example, to set the maximum frame rate to *60 frames per second*, set the maximum bandwidth to *900000 kilobits/second*, and the device bandwidth floor to *5000 kilobits/second*, you would set values in `pcoip-agent.conf` like this:

```
pcoip.maximum_frame_rate = 60
pcoip.max_link_rate = 900000
pcoip.device_bandwidth_floor = 5000
```

**Note: Support for IPv6 Addresses**

The Standard Agent for Linux supports IPv6 addresses. No configuration is needed to switch between IPv4 and IPv6 modes.

# H.264 Hardware Decode Requirements

For H.264 Hardware Decode, the **Graphics Agent** must have an NVIDIA Graphics Card that supports PCoIP Ultra GPU Offload, and PCoIP Ultra setting is either set to GPU Offload or Auto Offload.

**Note: Default Values**

In deployments with no existing PCoIP Ultra configuration, PCoIP Ultra defaults to "Auto Offload" and YUV Chroma subsampling defaults to 4:2:0.

The following NVIDIA graphics cards are supported:

- NVIDIA Quadro P400
- NVIDIA GeForce RTX 3060

Any Nvidia GPU with NVENC support are expected to work, but have not been tested.

**Note: Configurable Values**

A complete list of configurable values is shown next in Configurable Settings.

# Configurable Settings

The following settings can be configured on the Standard Agent for Linux. Refer to Configuring the Anyware agent to understand how to modify these settings.

## Authentication broker URL

| Directive | Options | Default |
|---|---|---|
| `pcoip.authentication_broker_url` | string *(up to **511** characters)* | — |

This setting takes effect when you start the next session. This policy sets the authentication broker URL for federated user authentication. Enter the authentication broker URL in 'https://address:port/auth' format. This setting overwrites the Authentication broker URL from Connection Manager.

# 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<br>**1**—Enabled in both directions<br>**2**—Enabled client to agent only<br>**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.

# Collaboration

| Directive | Options | Range | Increment | Default |
|---|---|---|---|---|
| `pcoip.enable_collaboration` | **0** (off), **1** (on) | | | Off |
| `pcoip.max_collaborators` | | 1 – 5 | 1 | 5 |
| `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 maximum number of collaborators allowed is 5.

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

If there is more than one collaborator, additional UDP ports will be needed for the collaborator sessions. For example, when the second collaborator connects, the next free UDP port will be opened on the host.

# Collaboration input control

| Directive | Options | Range | Increment | Default |
|---|---|---|---|---|
| `pcoip.enable_collaboration_input_control` | **0** (off),<br>**1** (on) | | | Off |
| `pcoip.collaboration_input_control_timeout` | | 100 – 10000 | 100 | 3000 |

This setting takes effect when you start the next PCoIP session. This policy enables or disables input control from the collaborators. When not configured, collaboration input control is disabled by default.

The input control timeout specifies the waiting period before any user with input control permission can acquire the input control of the host. The current input owner is the only one authorized to send mouse, keyboard and touch inputs to the host.

# 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.

# Desktop environment

| Directive | Options | Default |
|---|---|---|
| `pcoip.desktop_session` | string *(up to **511** characters)* | — |

This setting takes effect when you start the next session. Choose the desktop environment that will be launched from */usr/share/xsessions/*.desktop*, defaults to "kde-plasma" if present, else the first session found alphabetically in */usr/share/xsessions*.

Note that this setting only takes effect after an existing desktop session ends (either due to a reboot or logging out).

# Enable Disclaimer Authentication

| Directive | Options | Default |
|---|---|---|
| `pcoip.enable_disclaimer_auth` | **0** (off), **1** (on) | Off |

This setting takes effect when you start the next session. When this setting is enabled, users connecting via direct connect will be presented a disclaimer prior to user authentication. If the disclaimer is rejected, the user will not be able to connect.

Disclaimer files must be placed in */etc/pcoip-agent/disclaimers/* and must be readable by the "pcoip" system user. Files must be named according to the locale, e.g. en_US.txt for en_US, ko_KR.txt for ko_KR, etc. If a file matching the negotiated locale is not present, en_US will be used as a fallback. If disclaimer text cannot be found, an blank disclaimer will be presented.

## Enable/disable USB in the PCoIP session

| Directive | Options | Default |
|---|---|---|
| `pcoip.enable_usb` | **0** (off), **1** (on) | On |

This setting takes effect when you start the next session. Determines whether USB support is enabled in PCoIP sessions. When this setting is not configured, USB is enabled by default. By default all devices are supported unless restrictions are configured through the USB device rules setting.

## 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.

## Enable/disable relative mouse support

| Directive | Options | Default |
|---|---|---|
| `pcoip.enable_relative_mouse` | **0** (off), **1** (on) | On |

This setting takes effect when you start the next session. It determines whether relative mouse co-ordinates may be used, when appropriate, during the PCoIP session. By default, this setting is enabled.

# 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.

# Host key auto repeats

| Directive | Options | Default |
|---|---|---|
| `pcoip.use_host_autorepeat` | **0**—disabled (default)<br>**1**—use host generated auto repeats and client repeat rate (zero client only)<br>**2**—use host generated auto repeats and host repeat rate | — |

This setting takes effect when you start the next session. Configuring this allows you to enable or disable host generated key auto repeats. When not configured or disabled, key auto repeats are driven by the client.

# 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 Certificate Settings

| Directive | Options | Default |
|---|---|---|
| `pcoip.ssl_cert_type` | **1**—From certificate storage<br>**2**—Generate a unique self-signed certificate<br>**0**—From certificate storage if possible, otherwise generate | — |
| `pcoip.ssl_cert_min_key_length` | **1024**—1024 bits<br>**2048**—2048 bits<br>**3072**—3072 bits<br>**4096**—4096 bits | — |

This setting takes effect when you start the next session. A certificate is used to secure PCoIP related communications. The way PCoIP components choose a certificate is based on the certificate type and the key length. Without a certificate being generated or selected, a PCoIP Session cannot be established.

Depending on the value chosen for the option, 'How the PCoIP agent chooses the certificate...' and the availability of appropriate certificates, PCoIP components may acquire a CA signed certificate from certificate storage or generate an in-memory self-signed certificate.

In order for a CA signed certificate to be loadable by PCoIP components, it must be stored at */etc/pcoip-agent/ssl-certs* in three .pem files, owned by the pcoip user, only readable by the owning user.

- pcoip-key.pem must contain an unlocked RSA key
- pcoip-cert.pem must contain a certificate that signs the key in pcoip.pem
- pcoip-cacert.pem must contain a CA certificate chain that validates the certificate in pcoip-cert.pem.

Note: Self-signed certificates are 3072 bits long.

Select a minimum key length (in bits) for a CA signed certificate. Longer length certificates will require more computing resources and may reduce performance, but will increase security. Shorter length certificates will provide better performance at the cost of lower security.

Note: Please refer to Teradici documentation for instructions on creating and deploying certificates.

# PCoIP Security Settings

| Directive | Options | Default |
|---|---|---|
| `pcoip.tls_cipher_blacklist` | string *(up to* ***1023*** *characters)* | — |

This setting controls the cryptographic cipher suites used by PCoIP endpoints. Changes will take effect when the agent is restarted. When this setting is disabled or not configured, all supported cipher suites may be used for connections. The endpoints negotiate the actual cryptographic cipher suites based on the settings configured here. Newer versions of TLS and stronger cipher suites will be preferred during negotiation between endpoints. Supported cipher suites:

- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_AES_256_GCM_SHA384

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 USB allowed and unallowed device rules

| Directive | Options | Default |
|---|---|---|
| `pcoip.usb_auth_table` | **23XXXXXX**<br>**2203XXXX** | 23XXXXXX |
| `pcoip.usb_unauth_table` | **2203XXXX** | — |

This setting takes effect when you start the next session. This setting only applies if the related setting to Enable/disable USB in the PCoIP session is enabled, and specifies the USB devices that are authorized and not authorized for PCoIP sessions. Only devices listed in the USB authorization table are permitted in PCoIP sessions, provided they are not subsequently excluded by an entry in the USB unauthorization table.

You can define a maximum of 50 USB authorization rules and a maximum of 50 USB unauthorization rules. Separate multiple rules with the vertical bar (|) character. Please note the final number of authorization/unauthorization rules in a PCoIP session are negotiated by PCoIP client and agent. Some clients have a limit of 10 USB rules. Please refer to the PCoIP agent admin guide for details.

Each rule can be a combination of a Vendor ID (VID) and a Product ID (PID), or a rule can describe a class of USB devices. A class rule can allow or disallow an entire device class, a single subclass, or a protocol within a subclass.

The format of a combination VID/PID rule is 1xxxxyyyy, where xxxx is the VID in hexadecimal format and yyyy is the PID in hexadecimal format. For example, the rule to authorize or block a device with VID 0x1a2b and PID 0x3c4d is 11a2b3c4d.

For class rules, use one of the following formats:

Allow all USB Format: 23XXXXXX devices Example: 23XXXXXX

Allow USB Format: 22classXXXX devices with a Example: 22aaXXXX specific class ID

Allow a specific Format: 21class-subclassXX subclass Example: 21aabbXX

Allow a specific Format: 20class-subclass-protocol protocol Example: 20aabbcc

For example, the USB authorization string to allow USB HID (mouse and keyboard) devices (class ID 0x03) and mass storage devices (class ID 0x08) is 2203XXXX|2208XXXX. The USB unauthorization string to disallow USB Mass Storage devices (class ID 0x08) is 2208XXXX.

An empty USB authorization string means that no USB devices are authorized. An empty USB unauthorization string means that only USB devices in the authorization list are allowed.

If these settings are unconfigured, the default behavior is that all devices are allowed.

# PCoIP Ultra

| Directive | Options | Range | Increment | Default |
|---|---|---|---|---|
| `pcoip.enable_ultra` | **0** (off), **1** (on) | | | On |
| `pcoip.ultra` | **0**—Disabled<br>**1**—CPU Offload<br>**2**—GPU Offload<br>**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, 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 | — |
| `pcoip.yuv_chroma_subsampling` | **0**—4:4:4<br>**1**—4:2:0 | | | — |
| `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 4:2:0 by default. It enables chroma subsampling to further compress the imaging to reduce bandwidth usage at the cost of reduced color accuracy. 4:2:0 subsampling is only supported in combination with PCoIP Ultra GPU optimization. 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 fall back 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<br>**2**—Medium Priority (default)<br>**3**—Low Priority<br>**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.enable_vchan` | **1**—Enable all virtual channels other than those in the list<br>**2**—Disable all virtual channels other than those in the list | — |
| `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.

# Username comparison skipping

| Directive | Options | Default |
|---|---|---|
| `pcoip.skip_username_comparison` | **0** (off), **1** (on) | Off |

This setting takes effect when the agent is restarted. It allows username comparison to be skipped when launching the user's desktop environment. It should only be enabled when using a PAM stack where the desktop user may differ from the username used during login.

# X server remote access

| Directive | Options | Default |
|---|---|---|
| `pcoip.allow_x_remoting` | **0** (off), **1** (on) | Off |

This setting takes effect when you restart the agent. Configuring this allows you to enable or disable remote access to the X server run by the PCoIP Agent. When not configured, remote access is disabled by default.

# Making a Connection from a PCoIP Client

## Anyware Agent Deployment and Client Connectivity Requirements

Anyware 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 Anyware Architectures for each deployment's components and requirements.

⚠ **Connection troubleshooting**

If you encounter issues while connecting, see the Troubleshooting Connection Issues for fixes to common issues.

Once you've installed and configured your Standard Agent for Linux, you're ready to accept incoming connections from remote *Anyware 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 Standard Agent for Linux.

## Managing Client Connections

In most cases, Anyware clients connect to Anyware 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.

Anyware 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 Anyware 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.

## Anyware Manager

Anyware Manager is a service, available as a cloud-based service or as an installable instance, that centrally manages PCoIP deployments. It enables highly scalable and cost-effective HP Anyware deployments by managing cloud compute costs and brokering PCoIP connections to remote Windows or Linux workstations.

## Connection Manager

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

## Third-party Connection Brokers

Anyware agents also support third-party connection brokers. For a current list of brokering partners, see Technology Partners on the website.

# Additional Information

Information about **Anyware client connectivity requirements and usage instructions**, is available in the following documentation:

- Software clients:
  - Anyware Software Client for Windows
  - Anyware Software Client for macOS
  - Anyware Software Client for Linux

- Mobile Clients:
  - Anyware Mobile Client for iOS
  - Anyware Mobile Client for Android
  - Anyware Mobile Client for Chromebooks
- Zero clients:
  - Anyware Tera2 Anyware Zero Client

# Security Guide

PCoIP requires a certificate to establish a session. By default, Anyware 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 HP'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_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_AES_256_GCM_SHA384

**Note: Minimum SSL version**

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

# Custom Security Certificates

In order for a CA signed certificate to be loadable by Anyware components, it must be stored in `/etc/pcoip-agent/ssl-certs` in three **.pem** files, owned by the pcoip user, and only readable by the owning user:

- **pcoip-key.pem** must contain an unlocked RSA key
- **pcoip-cert.pem** must contain a certificate that signs the key in pcoip.pem
- **pcoip-cacert.pem** must contain a CA certificate chain that validates the certificate in pcoip-cert.pem

# Configure the Standard Agent for Linux to use custom certificates

The Standard Agent for Linux can be configured to look locally for certificates or to generate its own by setting the `pcoip.ssl_cert_type` directive in `pcoip-agent.conf`.

For more detailed information, see Configuring the Agent.

# Select a Security Key Length

When the Standard Agent for Linux is attempting to find a certificate in storage, the required key length can be set via the `pcoip.ssl_cert_min_key_length` directive in `pcoip-agent.conf`.

If the system cannot find a local certificate with the specified key length, it will either self-generate a certificate (if `pcoip.ssl_cert_type` is 0), or refuse the connection (if `pcoip.ssl_cert_type` is 1). This setting has no effect if `pcoip.ssl_cert_type` is set to 2.

For more detailed information, see Configuring the Agent.

# Reference

## Wacom Local Termination on Ubuntu Cloud Hosts

Cloud-based Ubuntu hosts will fail to properly identify Wacom tablets that have been locally-terminated at the Anyware client. When this occurs, pressure sensitivity and other advanced features will not work properly.

To work around this issue, remove the default AWS, Microsoft Azure, or Google Cloud kernel and replace it with a generic kernel.

> **Note: Ubuntu cloud hosts only**
>
> This procedure applies only to Ubuntu hosts on AWS, Microsoft Azure, or Google Cloud Platform. All valid RHEL and non-cloud Ubuntu installations work as expected.

**To enable local termination:**

1. First, confirm that you need to replace the kernel. Open a console and enter the following command:

   ```
   uname -r
   ```

   If the response contains the word `generic` (for example, `4.15.0-66-generic`) then your kernel is already generic and you can skip this procedure.

   If the response ends in `aws`, `azure`, or `gcp`, note the version number and continue.

2. Find the available `linux-virtual` package for your distribution. In a console window, enter the following command:

   ```
   apt-cache policy linux-virtual
   ```

   In the response, note the candidate major version number. For example, if the candidate's number is `4.15.0.66.68`, then the major version number is `4`.

3. Compare the major versions of the *installed* kernel from step 1, and the *candidate* kernel in step 2:

   - **If the major versions for the installed and candidate kernels are the same**

     In a console window, enter the following command:

     ```
     sudo apt install linux-virtual
     sudo apt install linux-cloud-tools-virtual
     ```

   - **If the major versions for the installed and candidate kernels are *not* the same**

     a. Retrieve the full list of available kernels:

     ```
     apt-cache policy linux-virtual*
     ```

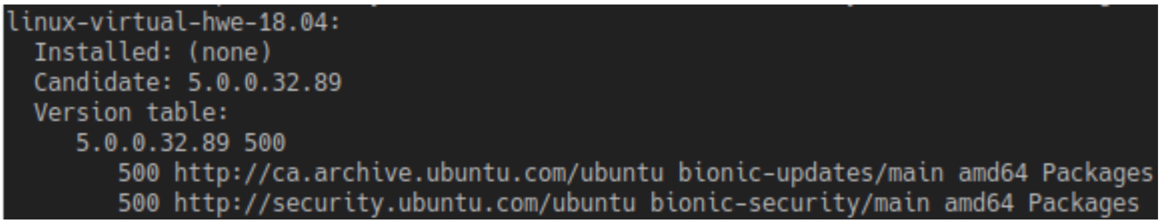
     Look through the output for the generic kernel version matching your installed kernel's major version.

     b. Install the kernel and cloud tools packages for the correct version:

     ```
     sudo apt install linux-virtual-<version>
     sudo apt install linux-cloud-tools-virtual-<version>
     ```

     ...where `<version>` is the number reported in the output from `apt-cache policy linux-virtual*`.

     For example, if you needed to find a kernel with a major version of `5`, you would look through the output of `apt-cache policy linux-virtual*` and find a response similar to this one:



```
linux-virtual-hwe-18.04:
  Installed: (none)
  Candidate: 5.0.0.32.89
  Version table:
     5.0.0.32.89 500
        500 http://ca.archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages
        500 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages
```

     The version is `hwe-18.04`. You will install that package and the corresponding cloud tools package:

     ```
     sudo apt install linux-virtual-hwe-18.04
     sudo apt install linux-cloud-tools-virtual-hwe-18.04
     ```

4. Purge the cloud-specific ubuntu image:

    - **AWS**:

      ```
      sudo apt purge linux*aws
      ```

    - **Azure**:

      ```
      sudo apt purge linux*azure
      ```

    - **GCP**

      ```
      sudo apt purge linux*gcp
      ```

5. When you see the *Abort kernel removal* message, respond with `No`.
6. Reboot the machine:

    ```
    sudo reboot
    ```

7. When the machine comes back up, reconnect and check that the generic kernel is in use:

    ```
    uname -r
    ```

    You should see a response ending in `-generic`.

8. Obtain the `uhid` driver by installing `linux-modules-extra` for your kernel version:

    ```
    sudo apt install linux-modules-extra-$(uname-r)
    ```

9. Reboot the machine:

    ```
    sudo reboot
    ```

10. When the machine comes back up, reconnect and check that the uhid driver is present:

    ```
    ls /dev/uhid
    ```

    You should see a response similar to `/dev/uhid`.

11. Install USB driver packages:

    ```
    sudo apt install usb-vhci-dkms
    ```

12. Reboot the machine:

    ```
    sudo reboot
    ```

13. When the machine comes back up, reconnect and check that the USB drivers are present:

    ```
    lsmod | grep usb
    ```

    You should see a response similar to this:

    ```
    usb_vhci_iocifc        20480  3
    usb_vhci_hcd           20480  1 usb_vhci_iocifc
    ```

    **Note: What if the response is empty?**

    If the output is empty, you may need to uninstall and reinstall the `vhci` package:

    ```
    sudo apt remove usb-vhci-dkms
    sudo reboot
    sudo apt install usb-vhci-dkms
    sudo reboot
    ```

14. Install the Wacom driver for your tablet.
15. Reboot the host machine.
16. If you have not installed the Standard Agent for Linux, install it now.

# Brokering Remote Workstation Card Machines

You can use the Standard Agent for Linux to provide brokering capabilities for your Linux Remote Workstation Card machines.

**Important**

Configuring your Anyware Zero Client's connection mode as described here will disable direct-to-host connections.

## Remote Workstation Card Desktop Requirements

The following requirements are specific to the Standard Agent for Linux when installed on Remote Workstation Card machines:

| Requirement | |
|---|---|
| Operating System | RHEL/CentOS 7.7 **only** |
| Remote Workstation Card Firmware | 5.1.0+ |
| Remote Workstation Card Software for Linux installed version | 4.8.0+ |

## Install the Standard Agent for Linux

Before you begin, confirm that your Remote Workstation Card and Remote Workstation Card Software are properly installed.

1. Confirm that you can create a direct connection from a Anyware Zero Client to the Remote Workstation Card machine. After verifying, disconnect the session.
2. Install the Standard Agent for Linux, using the procedure [here].

> **Important: Don't reboot yet**
>
> The installation procedure will tell you to reboot the machine in step 9; don't reboot it yet. We'll do that in a moment.

3. Open `/etc/pcoip-agent/pcoip-agent.conf` in a text editor.
4. Add the following line:

   ```
   pcoip.server_type = "RWC"
   ```

5. Save the file and close the editor.
6. Reboot the desktop.
7. Configure the Zero Client Session Connection as follows:
   - **Session Connection Type**: `PCM or AutoDetect`
   - **Server URI**: `<Host IP address or fqdn>`
8. Confirm your configuration by establishing a brokered connection.

# Networking

## IPv6

The Standard Agent for Linux supports IPv6 addresses. No configuration is needed to switch between IPv4 and IPv6 modes.

# Troubleshooting and Support

## Support

### Contacting Support

If you encounter any problems installing, configuring, or running the Standard Agent for Linux, you can create a support ticket.

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 local time when the problem occurred, in the HH:MM:SS format

#### The 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 PCoIP Technical Support Service team. The HP staff are heavily involved in the forums.

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

# Finding the Agent Version Number

**To find the agent's version number in Ubuntu**:

```
dpkg -l "pcoip*"
```

**To find the agent's version number in RHEL or CentOS**:

```
rpm -qai "pcoip*"
```

The console will display a table of all registered components and their version number, if they have one.

# Creating a Technical Support File

We may request a support file from your system in order to troubleshoot and diagnose issues. The support file is an archive containing Anyware Standard Agent for Linux logs and other diagnostic data that can help support diagnose your problem.

To create a support file, type the following command as a super user:

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

The support file will be created and placed in your `/tmp` directory. A message will display containing the full system path to the generated file.

# Troubleshooting

## Performing Diagnostics

Each Anyware component creates and updates a log file which records its activity as the system is used. Most troubleshooting within a Anyware 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 Standard Agent for Linux and other Anyware components are saved to specific directories.

**Note: Bundling log files for support**

When investigating issues with HP 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 Anyware 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` |
| Session Launcher | `/var/log/pcoip-agent/session-launcher.log` |
| Server/User | `/var/log/pcoip-agent/server.<user>.log` |

**Note: Bundling log files for support**

When investigating issues with HP support, you may need to provide a support file which includes system log files. Instructions are provided here.

## Setting Log Verbosity

Each Anyware 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 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 Anyware components.

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

## Log rotation

Log files in Linux agents are managed by `logrotate`. To manage how log files are rotated, edit the following files:

- `/etc/logrotate.d/pcoip-*`
- `/usr/share/pcoip-agent/pcoip-server.logrotate`

## Session Log IDs

At the start of each PCoIP session, a unique session ID is generated by the Anyware client and passed to all connected Anyware components (including the Standard Agent for Linux). 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 Anyware 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 Anyware 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

HP 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 HP Anyware 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

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

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

# Frequently Asked Questions

## Can I use a screensaver?

Yes. However, a blank, static screensaver will provide the most efficient CPU and network bandwidth usage.

## How quickly does a Anyware agent complete a connection?

Anyware agents can usually achieve a connection in 15 to 30 seconds. We use the statistical value Top Percentile (TP) to measure the time to establish a session:

- TP99: Ninety-nine percent of connections complete in under 30 seconds.
- TP50: Fifty percent of connections complete in under 15 seconds.

## Why is my application not sending audio?

The Anyware agent delivers audio over PCoIP connections by reassigning the system's default audio device. Only applications that use the system default audio device will send or receive audio over PCoIP; applications that are configured to use non-default devices will not work. If you don't hear audio from your application, make sure it is configured to use the system default audio device.

## I'm using Anyware Cloud Licensing. What network blocks should I leave open?

If you are using Anyware Cloud Licensing, you will need to add the following to your allowlist:

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

If you use an IP-based allowlist, we recommend your IT team add the following network blocks to your allowlist:

- `IPv4: 185.146.155.64/27`
- `IPv6: 2620:122:f005::/56`

**Important: Migrating from the previous specification**

Previously, our allowlist specification looked like this:

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

If you have an existing implementation using an IP-based allowlist like this, we recommend you leave it in place until the new allowlist is active and tested.