Working with Display Protocols
Contacting Leostream

Leostream Corporation
271 Waverley Oaks Rd.
Suite 206
Waltham, MA  02452
USA

http://www.leostream.com
Telephone: +1 781 890 2019

To submit an enhancement request, email features@leostream.com.
To request product information or inquire about our future direction, email sales@leostream.com.

Copyright

© Copyright 2002-2018 by Leostream Corporation

This software program and documentation are copyrighted by Leostream. The software described in this
document is provided under a license agreement and may be used or copied only under the terms of this
agreement. No part of this manual may be copied or reproduced in any form without prior written consent
from Leostream.

Trademarks

The following are trademarks of Leostream Corporation.
   Leostream™
   The Leostream graphical logo™

The absence of a product name or logo from this list does not constitute a waiver of the trademark or other
intellectual property rights concerning that product, name, or logo by Leostream.

HP is a registered trademark that belong to Hewlett-Packard Development Company, L.P. Oracle and Java
are registered trademarks of Oracle and/or its affiliates. Linux is the registered trademark of Linus Torvalds
in the U.S. and other countries. UNIX is a registered trademark of The Open Group. OpenLDAP is a
trademark of The OpenLDAP Foundation. Microsoft, Active Directory, SQL Server, Excel, ActiveX, Hyper-V,
Windows, and the Windows logo are trademarks or registered trademarks of Microsoft Corporation in the
United States and/or other countries. Other brand and product names are trademarks or registered
trademarks of their respective holders. Leostream claims no right to use of these marks.

Patents

Leostream software is protected by U.S. Patent 8,417,796.
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONTENTS</td>
<td>3</td>
</tr>
<tr>
<td>SUPPORTED DISPLAY PROTOCOLS</td>
<td>5</td>
</tr>
<tr>
<td>CHOOSING A DISPLAY PROTOCOL</td>
<td>5</td>
</tr>
<tr>
<td>CONFIGURING DISPLAY PROTOCOLS IN LEOSTREAM</td>
<td>7</td>
</tr>
<tr>
<td><strong>Using Protocol Plans</strong></td>
<td>7</td>
</tr>
<tr>
<td>How Protocol Plans Work</td>
<td>7</td>
</tr>
<tr>
<td>Building Protocol Plans</td>
<td>9</td>
</tr>
<tr>
<td><strong>Specifying Configuration Files and Command Line Arguments</strong></td>
<td>10</td>
</tr>
<tr>
<td>Using Dynamic Tags in Configuration Files</td>
<td>10</td>
</tr>
<tr>
<td>Dynamic Remapping of Desktop IP Address</td>
<td>14</td>
</tr>
<tr>
<td>Setting Configuration File Parameters Based on Client IP</td>
<td>15</td>
</tr>
<tr>
<td>EXCEED ONDEMAND</td>
<td>16</td>
</tr>
<tr>
<td>HP® REMOTE GRAPHICS SOFTWARE (RGS)</td>
<td>18</td>
</tr>
<tr>
<td>HP RGS Protocol Plan Options</td>
<td>18</td>
</tr>
<tr>
<td>Launching RGS Connections through the Leostream Gateway</td>
<td>19</td>
</tr>
<tr>
<td>Multi-Monitor Support with HP RGS</td>
<td>19</td>
</tr>
<tr>
<td>Activating HP Velocity and Advanced Video Compression Features</td>
<td>19</td>
</tr>
<tr>
<td>Setting User Configurable HP RGS Parameters</td>
<td>21</td>
</tr>
<tr>
<td>Single Sign-On with HP RGS</td>
<td>22</td>
</tr>
<tr>
<td>USB Passthrough with HP RGS</td>
<td>23</td>
</tr>
<tr>
<td>Session Shadowing and Collaboration</td>
<td>24</td>
</tr>
<tr>
<td>Using the HP RGS Receiver for Mac® OSX</td>
<td>24</td>
</tr>
<tr>
<td>Using HP SAM Clients</td>
<td>24</td>
</tr>
<tr>
<td>MICROSOFT® RDP AND REMOTEFX</td>
<td>26</td>
</tr>
<tr>
<td>Options for Encoding Desktop Login Credentials into RDP Configuration Files</td>
<td>26</td>
</tr>
<tr>
<td>Launching RDP Connections from the Leostream Web client</td>
<td>27</td>
</tr>
<tr>
<td>Microsoft RDP Viewer Command Line Parameters</td>
<td>27</td>
</tr>
<tr>
<td>Microsoft RDP Viewer Configuration File Variables</td>
<td>28</td>
</tr>
<tr>
<td>Connecting to RemoteApp Servers</td>
<td>35</td>
</tr>
<tr>
<td>Integrating with a Microsoft Remote Desktop Gateway</td>
<td>36</td>
</tr>
<tr>
<td>MECHDYNE TGX</td>
<td>37</td>
</tr>
<tr>
<td>Launching Mechdyne TGX Connections</td>
<td>37</td>
</tr>
<tr>
<td>Setting User-configurable TGX Parameters</td>
<td>38</td>
</tr>
<tr>
<td>NICE DESKTOP CLOUD VISUALIZATION (DCV)</td>
<td>40</td>
</tr>
<tr>
<td>Launching NICE DCV Connections</td>
<td>40</td>
</tr>
<tr>
<td>NOMACHINE</td>
<td>41</td>
</tr>
<tr>
<td>Launching NoMachine Connections from the Web Client</td>
<td>41</td>
</tr>
<tr>
<td>NoMachine Configuration File</td>
<td>42</td>
</tr>
<tr>
<td>Session Shadowing and Collaboration</td>
<td>42</td>
</tr>
<tr>
<td>Setting User-configurable NoMachine Parameters</td>
<td>42</td>
</tr>
<tr>
<td>Topic</td>
<td>Page</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>TERADICI® PCOIP® TECHNOLOGY</td>
<td>45</td>
</tr>
<tr>
<td><strong>Using PCoIP Clients with Leostream</strong></td>
<td>45</td>
</tr>
<tr>
<td><strong>Enabling PCoIP Remote Workstation Card Support</strong></td>
<td>46</td>
</tr>
<tr>
<td><strong>Connecting to PCoIP Remote Workstation Cards</strong></td>
<td>46</td>
</tr>
<tr>
<td><strong>Using the Teradici Cloud Access Software and Cloud Access Platform</strong></td>
<td>47</td>
</tr>
<tr>
<td><strong>PCoIP Connections to VMware Virtual Machines with a View Direct-Connection Plug-In</strong></td>
<td>47</td>
</tr>
<tr>
<td>Establishing Connections using Leostream Connect</td>
<td>48</td>
</tr>
<tr>
<td>Establishing Connections using the Leostream Web Client</td>
<td>49</td>
</tr>
<tr>
<td>Establishing Connections using a PCoIP Zero Client</td>
<td>49</td>
</tr>
<tr>
<td>RDHTMLP RDP REMOTE VIEWER</td>
<td>51</td>
</tr>
<tr>
<td>VNC REMOTE VIEWER</td>
<td>51</td>
</tr>
<tr>
<td>Setting up the Connection Broker to Use VNC</td>
<td>51</td>
</tr>
<tr>
<td>VNC Command Line Parameters</td>
<td>52</td>
</tr>
<tr>
<td>SESSION SHADOWING AND COLLABORATION</td>
<td>55</td>
</tr>
<tr>
<td>Configuring Collaboration in the Connection Broker</td>
<td>55</td>
</tr>
<tr>
<td>Working with Invitations in the Leostream Web Client</td>
<td>56</td>
</tr>
<tr>
<td>Sending a Collaboration Invitation</td>
<td>56</td>
</tr>
<tr>
<td>Cancelling an Invitation</td>
<td>58</td>
</tr>
<tr>
<td>Accepting a Collaboration Invitation</td>
<td>59</td>
</tr>
<tr>
<td>Working with Invitations using Leostream Connect</td>
<td>59</td>
</tr>
<tr>
<td>Sending a Collaboration Invitation</td>
<td>59</td>
</tr>
<tr>
<td>Viewing and Cancelling Invitations</td>
<td>60</td>
</tr>
<tr>
<td>Accepting a Collaboration Invitation</td>
<td>60</td>
</tr>
<tr>
<td>Managing Invitations in the Connection Broker</td>
<td>60</td>
</tr>
<tr>
<td>USER CONFIGURABLE PROTOCOL PLAN PARAMETERS</td>
<td>62</td>
</tr>
<tr>
<td>Defining Scope of the Configured Parameter</td>
<td>62</td>
</tr>
<tr>
<td>End-User Interface for Configuring Parameters</td>
<td>63</td>
</tr>
<tr>
<td>Leostream Web Client</td>
<td>63</td>
</tr>
<tr>
<td>Leostream Connect</td>
<td>64</td>
</tr>
<tr>
<td>Setting Global User-Configurable Parameters</td>
<td>64</td>
</tr>
</tbody>
</table>
Supported Display Protocols

The Leostream Connection Broker supports a wide range of display protocols that allow you to provide the required end-user experience throughout your entire organization. The Leostream Connection Broker can launch desktop connections using the following display protocols.

- HP Remote Graphics Software (RGS)
- Leostream HTML5-based RDP, VNC, and SSH
- Microsoft RDP and RemoteFX
- Mechdyne TGX
- NICE Desktop Cloud Visualization (DCV)
- NoMachine and FreeNX
- OpenText Exceed onDemand
- rdesktop
- Teradici PCoIP
- VNC (RealVNC, TigerVNC, TightVNC, and UltraVNC)

Connection Broker protocol plans define which display protocols are used, and how the remote session is launched. Defining protocol plans is covered in Configuring Display Protocols in Leostream.

Before you build your protocol plans, you must choose the display protocols you will use in your environment. The next chapter provides general guidelines when considering different display protocols.

Choosing a Display Protocol

Leostream can establish a connection to a remote desktop using a variety of supported display protocols. After the connection is established, the Connection Broker removes itself from the connection path, i.e., the Connection Broker does not proxy the remote session.

The performance and requirements of the remote session are, therefore, completely determined by the display protocol you select. This chapter provides some food-for-thought when investigating and choosing from the available display protocols.

Choosing the right protocol requires a balance between good end-user experience, the bandwidth available on the network, and the compute power supplied by the hardware. Every display protocol balances these requirements, with the ultimate goal being:

- Low bandwidth requirement
- Low computational requirements
- High-quality end-user experience

These three factors make up the protocol triangle, depicted in the following figure. As with any triangle, changing the angle for one corner has repercussions for the other angles.
You can typically achieve any two of your display protocol goals, but will have to compromise on the third. For example, if your users’ needs are met with a lower performance viewing experience, you can choose a protocol that requires lower bandwidth and lower computing power. However, if you must provide a high-performance viewing experience, you need either higher bandwidth or higher computing power, and ideally both.

Each available display protocol handles the corners of the protocol triangle differently; each has its benefits and its drawbacks. When picking one or more display protocols, determine which protocol characteristics you need, and which trade-offs you can accept.

The following questions may help you define your display protocol requirements. Outline as many requirements as possible, and test different display protocols in your environment before committing to a particular protocol. Leostream can leverage multiple protocols in a single environment, so do not feel like you must commit to a single display protocol.

- What are your end users’ requirements for multi-media, USB device redirection, response time, etc.?

- Do you have different types of users, for example task workers that run word processing applications and power users running graphic-intensive applications?

- What operating systems are you planning to deliver on your remote desktops, or use on your client devices? Are you planning to support BYOD? If so, make sure your chosen display protocol handles all possible client devices.

- Do you want to use zero or thin clients? If you are using zero clients, which display protocol does it natively support? If using thin clients, what operating system does it use and can you install additional client software?

- Are your users accessing an entire desktop or only an application?

- Is single sign-on a requirement, or just nice-to-have?

- Do you need a display protocol that supports collaboration, where two users are simultaneously logged into the same session?

- How large will your deployment grow? (High compute power requirements affect scalability.)
Working with Display Protocols

- Do users connect to workstation/ blades with native compute power, or are you using a virtual environment that supports GPU passthrough or vGPU?

Configuring Display Protocols in Leostream

Using Protocol Plans

Connection Broker protocol plans define which display protocol the Connection Broker uses when connecting a user to their desktop. Protocol plans define the order in which the Connection Broker tries to use the available protocols when connecting to a desktop, and the configuration file or command line parameters used for the connection.

The Connection Broker provides one default protocol plan, which is shown on the > Configuration > Protocol Plans page, shown in the following figure.

Each protocol plan defines the display protocol used when the user logs in from Leostream Connect and thin clients, Dell Wyse® ThinOS clients, the Leostream Web client, PCoIP clients, and HP SAM clients. You configure the display protocol for each of these client types separately, using the appropriate section in the protocol plan.

⚠️ Your Leostream license determines which display protocols are included in your Connection Broker. Please, contact sales@leostream.com if you need access to additional display protocols.

How Protocol Plans Work

Protocol plans give you the flexibility to configure which display protocol to use for each pool used in a policy. A protocol plan tells the Connection Broker:

- Which display protocols are allowed for this pool

- What priority each protocol has, i.e., which protocol should the Connection Broker try first, second, etc.
What, if any, command line parameters and configuration file should the Connection Broker use when establishing the connection?

The following figure shows a portion of the *Leostream Connect and Thin Clients Writing to Leostream API* section of a protocol plan.

The selection in the **Priority** drop-down menu indicates the order in which the Connection Broker tries to establish a connection using that display protocol. In the previous figure, the Connection Broker first tries Microsoft RDP, which has a priority of 1. If an RDP connection cannot be established, the Connection Broker then tries HP RGS, which has a priority of 2. If HP RGS also fails, the Connection Broker looks for a protocol with a **Priority** of 3. If the **Priority** drop-down menu for all other display protocols is set to **Do not use**, the Connection Broker returns a warning and does not establish a connection to the remote desktop.

To determine if a particular display protocol can be used, the Connection Broker performs a port check on the remote desktop. For example, by default, Microsoft RDP communicates over port 3389. For the above example, if port 3389 is open on the remote desktop, the Connection Broker connects to the desktop using RDP. If port 3389 is not open, the Connection Broker checks the default RGS Sender port 42966.

⚠️ The Connection Broker cannot distinguish between display protocols that use the same port, for example Microsoft RDP and rdesktop. Therefore, if a protocol plan sets the priority for Microsoft RDP to 1, and the priority of rdesktop to 2, the Connection Broker always uses RDP when port 3389 is open on the remote desktop.
remote desktop, even if you are connecting from a Linux client that supports only rdesktop. For this example, you need a second protocol plan that assigns a priority of 1 to rdesktop, to support users logging in from a Linux client.

The Connection Broker does not consider the client device’s capabilities when choosing the display protocol. For example, if the protocol plan sets the priority of HP RGS to 1 and the port check passes on the remote desktop, the Connection Broker instructs the client device to launch an RGS connection, even if the client does not have an installed RGS Receiver.

**Building Protocol Plans**

To determine how many protocol plans you need, and how they should be configured, think about all the different ways your end users will connect to their desktops, for example:

- Do all users access their desktops using the same display protocol? If not, which protocols will they use? If these protocols communicate over the same port, you will need a protocol plan for each protocol.

- For each display protocol that you use, will the command line parameters and configuration file be the same for all users? If not, you will need a protocol plan for each configuration.

- Do your remote desktops support multiple protocols, such as RDP, RGS, and VNC? If so, and different users will launch connections with different protocols, you need a protocol plan that defines the appropriate priorities for each remote viewer.

The above questions are examples of things you should think about when building protocol plans. Begin with a simple scenario and create your protocol plan as follows.

1. Go to the > **Configuration > Protocol Plans** page.

2. Click the **Create Protocol Plan** at the top of the page. The **Create Protocol Plan** form opens.

3. In the **Plan name** edit field, enter the name to use when referring to this protocol plan.

4. In the **Leostream Connect and Thin Clients Writing to Leostream API** section, shown in the previous figure, configure the protocols to use when a user logs in using one of the following client devices:

   - The Windows or Java version of Leostream Connect, installed on a Windows, Linux, or Mac OSX device.
   - A thin client with an installed Leostream Connect client
   - A thin client with a customized Leostream client

5. In the **Web Browser** section, shown in the following figure, configure the protocol to use when a user logs in through the Leostream Web client.
6. Configure the remainder of the protocol plan if your end users log in through any of the following client devices.

- Wyse thin clients running the Wyse Thin OS
- PCoIP clients
- HP SAM clients

7. Use the Notes field to store any additional information with your protocol plan.

8. Click Save to store any changes to the plan.

### Specifying Configuration Files and Command Line Arguments

Configuration files and command line parameters allow you to customize the remote session. The format and contents of these fields differs for each display protocol. The following chapters discuss each display protocol and provide some example syntax. The remainder of this chapter discusses Connection Broker concepts pertaining to using dynamic tags in a configuration file or command line parameter.

### Using Dynamic Tags in Configuration Files

Configuration files and command line parameters allow you to customize how the display protocol’s client software establishes the desktop connection, for example, if the connection opens in full screen or windowed mode. The Connection Broker supports dynamic tags in the Command line parameters and Configuration file fields for any of the protocol. Before passing the command line parameters or configuration file to the client device, the Connection Broker replaces dynamic tags with the appropriate information, allowing you to reuse protocol plans for multiple users.

The following table contains a complete list of the supported dynamic tags. If the configuration file contains...
text enclosed in braces that is not included in the list of supported dynamic tags, the Connection Broker ignores the tag and it remains in the configuration file.

<table>
<thead>
<tr>
<th>Dynamic Tags</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>{IP}</td>
<td>The IP address of the Leostream Agent on the desktop. If no Leostream Agent is installed on the desktop, {IP} is replaced with the hostname of the desktop or, if the hostname is not available, the IP address of the desktop.</td>
</tr>
<tr>
<td>{IP_PRIVATE}</td>
<td>For desktops hosted in OpenStack, AWS, and Azure, the internal IP address seen by the operating system.</td>
</tr>
<tr>
<td>{IP_PUBLIC}</td>
<td>For desktops hosted in OpenStack, AWS, and Azure, the external IP address, if allocated, that is accessible from the outside network.</td>
</tr>
<tr>
<td>{IP_ADDRESS}</td>
<td>The IP address of the desktop.</td>
</tr>
<tr>
<td>{HOSTNAME}</td>
<td>The hostname of the desktop.</td>
</tr>
<tr>
<td>{IP_ADDRESS-or-HOSTNAME}</td>
<td>The IP address of the desktop or, if the IP address is not available, the hostname of the desktop.</td>
</tr>
<tr>
<td>{HOSTNAME-or-IP_ADDRESS}</td>
<td>The hostname of the desktop or, if the hostname is not available, the IP address of the desktop.</td>
</tr>
<tr>
<td>{SHORT_HOSTNAME}</td>
<td>The short hostname of the desktop, or the hostname cut at the first dot. For example, if the hostname is desktop.example.com, the {SHORT_HOSTNAME} tag returns desktop.</td>
</tr>
<tr>
<td>{DCV_PORT}, {VNC_PORT}</td>
<td>For DCV and VNC connections, the port for the VNC session, as returned by the Leostream Agent.</td>
</tr>
<tr>
<td>{USER},{USER:USER},{USER:LOGIN_NAME}, or {LOGIN)NAME}</td>
<td>The user's login name. This value corresponds to the value shown in the Login name column on the &gt; Resources &gt; Users page. To force the login name on the remote desktop to upper or lower case, include the :lowercase or :uppercase modifier, for example {USER:lowercase} or {USER:LOGIN_NAME:uppercase}.</td>
</tr>
<tr>
<td>{AD:USER:attribute_name}</td>
<td>The value found in the user's Active Directory attribute given by attribute_name. Use this dynamic tag if you need to replace the user's login name for their remote session with a value different from the login name used for their Leostream session. See “Using Dynamic Tags” in the Connection Broker Administrator's Guide.</td>
</tr>
<tr>
<td>{NAME} or {USER:NAME}</td>
<td>The user's display name. This value corresponds to the value shown in the Name column on the &gt; Resources &gt; Users page.</td>
</tr>
<tr>
<td>{AD_DN} or {USER:AD_DN}</td>
<td>The user's Active Directory Distinguished Name. This value corresponds to the value shown in the AD Distinguished Name column on the &gt; Resources &gt; Users page.</td>
</tr>
<tr>
<td>Dynamic Tags</td>
<td>Purpose</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>{EMAIL} or {USER:EMAIL}</td>
<td>The user's email address. This value corresponds to the value shown in the Email column on the Resources &gt; Users page.</td>
</tr>
<tr>
<td>{PRE_EMAIL} or {USER:PRE_EMAIL}</td>
<td>The portion of the user's email address before the @ symbol.</td>
</tr>
<tr>
<td>{POST_EMAIL} or {USER:POST_EMAIL}</td>
<td>The portion of the user's email address after the @ symbol.</td>
</tr>
<tr>
<td>{DOMAIN}</td>
<td>The name entered into the Domain field for the authentication server that authenticated a user. If the Domain field is empty, the Connection Broker replaces this dynamic tag with the value entered or selected in the Domain field of the user's client.</td>
</tr>
<tr>
<td>{AUTH_DOMAIN}</td>
<td>The name entered in the Authentication server name field of the authentication server that authenticated the current user.</td>
</tr>
<tr>
<td>{PLAIN_PASSWORD}</td>
<td>The user's password, in plain text</td>
</tr>
<tr>
<td>{RDP_PASSWORD}</td>
<td>For Leostream Connect, the user’s password encrypted for RDP usage</td>
</tr>
<tr>
<td>{SCRAMbled_PASSWORD}</td>
<td>For NoMachine, only, the user's password scrambled to prevent casual eavesdropping.</td>
</tr>
<tr>
<td>{CREDENTIALS_MECHDYNE}</td>
<td>Encrypted user credentials to pass to the TGX Sender to provide single sign-on.</td>
</tr>
<tr>
<td>{STANDARD_RDP_PASSWORD:xxxx}</td>
<td>For Leostream Connect, a specific password encrypted for RDP usage</td>
</tr>
<tr>
<td>{CLIENT} or {CLIENT:NAME}</td>
<td>The name of the user's client device used to log into the Connection Broker. This value corresponds to the value shown in the Name column on the Resources &gt; Clients page.</td>
</tr>
<tr>
<td>{CLIENT:IP}</td>
<td>The IP address of the user's client device used to log into the Connection Broker. This value corresponds to the value shown in the IP Address column on the Resources &gt; Clients page.</td>
</tr>
<tr>
<td>{CLIENT:MAC}</td>
<td>The MAC address of the user's client device used to log into the Connection Broker. This value corresponds to the value shown in the MAC Address column on the Resources &gt; Clients page.</td>
</tr>
<tr>
<td>{CLIENT:TYPE}</td>
<td>The type of client used to log into the Connection Broker. This value corresponds to the value shown in the Type column on the Resources &gt; Clients page.</td>
</tr>
<tr>
<td>{CLIENT:MANUFACTURER}</td>
<td>The manufacturer of client used to log into the Connection Broker. This value corresponds to the value shown in the Manufacturer column on the Resources &gt; Clients page.</td>
</tr>
<tr>
<td>{CLIENT:UUID}</td>
<td>The UUID of the client used to log into the Connection Broker. This value corresponds to the value shown for the Client UUID on the Edit Client page.</td>
</tr>
<tr>
<td>Dynamic Tags</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>{PCOIP_HOST1}</code> or <code>{PCOIP_HOST2}</code></td>
<td>The last know IP address of the Teradici PCoIP Remote Workstation Card associated with the desktop for the connection. If the Connection Broker does not have an IP address for the card, then the dynamic tag is replaced with the card’s hostname.</td>
</tr>
<tr>
<td><code>{POOL:NAME}</code></td>
<td>The name of the pool that contains the desktop that the user is connecting to</td>
</tr>
<tr>
<td><code>{VM:NAME}</code></td>
<td>The name of the desktop the user is connecting to, as shown in the Name field on the &gt; Resources &gt; Desktops page.</td>
</tr>
<tr>
<td><code>{WINDOWS_NAME}</code></td>
<td>The guest host name of a desktop, as returned by the Leostream Agent</td>
</tr>
<tr>
<td><code>{FQDN}</code></td>
<td>If the user authenticated against an authentication server, the fully qualified name, e.g., cn=Fred,ou=Users,o=Company</td>
</tr>
<tr>
<td><code>{DRIVE:CD}</code></td>
<td>For the RDP configuration file, use drivestoredirect:s:{DRIVE:CD} to redirect all CD drives found on system. No other drives are directed.</td>
</tr>
<tr>
<td><code>{DRIVE:DVD}</code></td>
<td>For the RDP configuration file, use drivestoredirect:s:{DRIVE:DVD} to redirect all DVD drives found on system. No other drives are directed.</td>
</tr>
<tr>
<td><code>{LEO_SPAN}</code></td>
<td>For use with display plans, either 1 or 0 depending on if the RDP session should be spanned across multiple monitors.</td>
</tr>
<tr>
<td><code>{LOGOUT_URL}</code></td>
<td>The URL to log the user out of the session.</td>
</tr>
<tr>
<td><code>{LIST_URL}</code></td>
<td>The URL to view the list of desktops.</td>
</tr>
<tr>
<td><code>{ENV:*}</code></td>
<td>The value of the client side variable specified in *. So <code>{ENV: HTTP_COOKIE}</code> might return <code>uid=25157202</code>.</td>
</tr>
<tr>
<td><code>{REMAPPED_IP:X.X.X.X}</code></td>
<td>Re-maps IP addresses by replacing the non-X portion of the IP address with the specified tag.</td>
</tr>
<tr>
<td><code>{REMAPPED_IP:subnet_mask}</code></td>
<td>Re-maps IP addresses on different subnets.</td>
</tr>
<tr>
<td><code>{SESSION}</code></td>
<td>For use with the Java version of Leostream Connect. The session ID associated with session-based RGS Receiver configuration file parameters.</td>
</tr>
<tr>
<td><code>{USB_SESSION}</code></td>
<td>Indicates that the Java version of Leostream Connect should manage which remote RGS session has access to USB devices.</td>
</tr>
</tbody>
</table>
Dynamic Tags | Purpose
---|---
{MATCHED_IP:partial_IP_address} | The IP address of the desktop, where *partial_IP_address* indicates that the Connection Broker should favor IP addresses that begin with the specified values. Typically, when a desktop has multiple network interfaces, the Leostream Agent and Connection Broker negotiate which IP address to use for remote connections. By using the MATCHED_IP dynamic tag, you instruct the protocol plan to favor a specific IP address. For example, if the desktop returns two IP addresses of 172.29.229.151 and 10.110.1.14 and the tag is {MATCHED_IP:10.110.1} the IP address used for the connection is 10.110.1.14.

If the desktop does not have an IP address beginning with the values to match, the Connection Broker will not establish a remote connection to the desktop. To allow the Connection Broker to fail over to another IP address, use the syntax {MATCHED_IP:partial_IP_address-or-IP}. If the desktop returns one IP addresses of 172.29.229.151 and the tag is {MATCHED_IP:10.110.1-or-IP} the IP address used for the connection is 172.29.229.151.

When specifying *partial_IP_address*, trailing zeros are optional, e.g., {MATCHED_IP:172.29.0.0} is equivalent to {MATCHED_IP:172.29}.

### Dynamic Remapping of Desktop IP Address

You can enable display protocol traffic to traverse one or more NAT firewalls by dynamically changing the IP address provided to the display protocol’s client to reflect the address of the desktop seen from the client’s perspective as opposed to that seen by the desktop.

To do this, use the {REMAPPED_IP} dynamic tag in place of the {IP} dynamic tag. The Connection Broker applies the IP address mask specified in the dynamic tag to the IP address of the desktop, so that the address is modified.

As an example, imagine an offshore development center that runs on a 192.168.1.xxx network. One of its customers has a series of desktops running on a 172.29.229.xxx network. A NAT firewall makes the transition between the two networks. Therefore, a desktop at 172.29.229.131 appears to the offshore development center as a desktop at 192.168.1.131.

To accomplish this transition, in the configuration file, change instances of the {IP} tag to {REMAPPED_IP:192.168.1.X}.

To remap IP addresses on multiple subnets, use the advanced form of the {REMAPPED_IP} dynamic tag.
This version of the `{REMAPPED_IP}` dynamic tag specifies a network mask length and a target range for the source and destination.

Use the `{REMAPPED_IP:X.X.X.X}` syntax to perform DNS resolution without remapping the IP address.

Use the wildcard (*) to map all subnets. For example:

- `{REMAPPED_IP:*/24->192.168.1.0}` replaces the first 24 bits of the IP address on all subnets with 192.168.1. Therefore, the IP address 10.153.172.5 maps to 192.168.1.5.

- `{REMAPPED_IP:*/8->194.0.0.0}` replaces the first 8 bits of the IP address on all subnets with 194. Therefore, the IP address 10.153.174.9 maps to 194.153.174.9.

To map different subnets to different IP address ranges, use the syntax in the following example.

```
{REMAPPED_IP:10.153.174.0/24 -> 192.168.204.0, 10.153.172.0/24 -> 192.168.201.0}
```

Each subnet map is separated by a comma. A subnet map can be defined using a wildcard, as described in the earlier `{REMAPPED_IP}` examples.

In this example, the first 24 bits of IP addresses in the subnet 10.153.174 are mapped to 192.168.204, while the first 24 bits of the IP addresses in the subnet 10.153.172 are mapped to 192.168.201. Therefore:

- 10.153.174.9 maps to 192.168.204.9
- 10.153.172.5 maps to 192.168.201.5
- 10.153.173.7 remains 10.153.173.7

In cases where multiple subnet maps are included, the order of the maps is irrelevant. The more specific map takes precedence over the less specific map. When a wildcard is provided, any IP addresses that are not mapped by one of the other rules will be mapped by the wildcard. The Connection Broker always performs wildcard mappings last.

Do not specify multiple wildcard mappings. If multiple wildcards are specified, the Connection Broker uses one of the mappings and ignores all other maps.

**Setting Configuration File Parameters Based on Client IP**

You can use the client’s IP address to determine if a particular client configuration variable is sent to the client. You can also differentiate between users connecting locally and the same user connecting remotely.

To do this, wrap the configuration file parameter in a logic statement. These logic statements contain three parts. They can all be together, or on different lines. For example:
Working with Display Protocols

{NETWORK:10.0.0.0/255.255.255.0}
  compression:1
{END_NETWORK}

- {NETWORK:10.0.0.0/255.255.255.0} defines a network IP address range. The client IP address must fall within this range for the logic statement to be true.

- compression:1 defines the configuration setting to be used if the logic statement is true. You can include multiple parameters spread across multiple lines.

- {END_NETWORK} closes the logic statement and must be present.

If you use multiple logic statements, the Connection Broker evaluates the statements in order and includes the parameter for the first statement that evaluates to true. For example:

{NETWORK:10.0.0.0/255.255.255.0} compression:0{END_NETWORK}
{NETWORK:192.168.10.0/255.255.255.128} compression:1{END_NETWORK}
{NETWORK:0.0.0.0/0.0.0.0}
  compression:0
{END_NETWORK}

In this example, if the client's IP address is 10.0.0.* they have data compression turned off. If the address is between 192.168.10.1 and 192.168.10.127 (VPN connected users), compression is turned on. If the address is anything else, compression is turned off.

Exceed OnDemand

Exceed onDemand from OpenText provides pixel perfect screen and color rendering for professionals in design and manufacturing industries, allowing organizations to deliver complex 2-D and 3-D applications to a global work force with LAN-like performance. For more information on Exceed onDemand, visit the OpenText Web site.

⚠️ Leostream does not currently support Exceed TurboX. Please, contact sales@leostream.com to request this functionality.

The Leostream clients that support Exceed onDemand include:

- Leostream Connect for Windows and Linux operating systems
- The Leostream Web client

To configure the Connection Broker to request an Exceed onDemand connection:

1. Go to the > Configuration > Protocol Plans page

2. Click the Create Protocol Plan link, to build a new protocol plan for Exceed onDemand

3. In the Leostream Connect and Web Browser sections of the protocol plan form, switch the Priority
for RDP to **Do not use**.

4. Switch the **Priority** of **Exceed onDemand** to 1.

5. For Leostream Connect, in the **Command line parameters** field, enter any options to pass to the Exceed onDemand client. See the Exceed onDemand Client documentation for a list of supported command line parameters.

   Enter `-e` if users are allowed to select their `Xconfig` and `Xstart` file after connecting to the Exceed onDemand server.

6. In the **Configuration file** field, enter the EOD file to use to launch the connection. You can obtain a default EOD file by saving a connection document from within the Exceed onDemand client.

7. Also in the **Configuration file** field, use the following dynamic tags in the EOD file, to support multiple users from the same protocol plan:

   - `SERVER={IP}`
   - `UID={USER}`
   - `PASSWD={SCRAMBLED_PASSWORD}`

   For example, your configuration file may look something like this:

   ```
   [EOD8CONNECTION]
   [EODCONNECTION]
   VERSION=13.8.2.450
   SERVER={IP}
   PORT=5500
   UID={USER}
   PASSWD={SCRAMBLED_PASSWORD}
   XSTART=-;X_Window_Manager.xs
   ISGLOBALXS=1
   PROFILE=-;Passive.cfg
   ISGLOBALPROF=1
   ```

8. **Click Save**.

The Leostream Agent tracks Exceed sessions, as follows.

- Starting an Exceed session maps to a Login event
- Suspending an Exceed session maps to a Disconnect event
- Resuming an Exceed session maps to a Connect event
- Closing an Exceed session maps to a logout

⚠️ If a user logs out of, for example, a gnome desktop session but leaves the Exceed connection open, the Connection Broker does not receive a logout event from the Leostream Agent. The user must close the Exceed session to register a logout event.
• Users connecting to an Exceed server that do not log in via Leostream Connect or the Leostream Web client are reported as rogue users. The exception is users who shadow another user’s Exceed session. The Leostream Agent recognizes users who log in as a shadow and does not report them as rogue.

HP® Remote Graphics Software (RGS)

HP® Remote Graphics Software (RGS) is a high performance remote graphics system that renders the graphics on the desktop and sends the resulting screen image to the remote client. To learn more about HP RGS, refer to the HP RGS product page.

Leostream Connect has been fully qualified with HP RGS version 6.x, and 7.x.

⚠️ RGS 7.0 and 7.0.2 have known issues when used with Leostream. For a Windows operating system, you must use RGS Sender version 7.0.1, or higher. For a Linux operating system, use RGS Sender version 7.0.1, only. RGS Sender version 7.0.2 for Linux has known issue when used with the Connection Broker.

HP RGS Protocol Plan Options

The HP RGS section of protocol plans, shown in the following figure, allows you to specify the user login name and RGS Receiver parameters used to launch the connection. To ensure that the Connection Broker establishes an HP RGS connection, switch the Priority for RGS to 1.

Edit the Send user login name as edit field if you need to use a login name for the RGS session that is different from the login name used for the Leostream session. You can use any of the dynamic tags associated with the user’s account described in Using Dynamic Tags in Configuration Files, including {USER}, {EMAIL}, {AD:USER:attribute_name}, etc.

Use the Configuration file field associated with RGS to specify RGS Receiver properties, as shown in the following figure. The text you enter into the Configuration file field is analogous to the rgreceiverconfig file that sets RGS Receiver parameters on the client computer when making native HP RGS connections to a remote desktop. The default configuration file is:
Working with Display Protocols

Rgreceiver.IsBordersEnabled=0
Rgreceiver.IsBordersEnabled.IsMutable=0
Rgreceiver.IsMatchReceiverResolutionEnabled=1
Rgreceiver.IsMatchReceiverResolutionEnabled.IsMutable=0
Rgreceiver.IsMatchReceiverPhysicalDisplaysEnabled=1
Rgreceiver.IsMatchReceiverPhysicalDisplaysEnabled.IsMutable=0

See the HP Remote Graphics Software User Guide for a complete description of the available RGS Receiver properties. Every RGS Receiver installation provides a documented example rgreceiverconfig file in the installation directory, for example:

```
C:\Program Files\Hewlett-Packard\Remote Graphics Receiver\rgreceiverconfig
```

The Connection Broker does not provide separate command line parameters for RGS. All command line parameters must be set using their configuration file equivalents.

**Launching RGS Connections through the Leostream Gateway**

The Leostream Gateway can proxy HP RGS connections, allowing you to establish RGS connections from clients on different networks than the RGS Sender. For a complete description of using the Leostream Gateway for RGS connections, consult the Leostream Gateway Guide.

**Multi-Monitor Support with HP RGS**

The HP RGS Sender can automatically change the display settings on the remote desktop to match the monitor layout and resolution used on the client device running the RGS Receiver.

The default configuration file in new protocol plans enables the
IssuerMatchReceiverResolutionEnabled and
IssuerMatchReceiverPhysicalDisplaysEnabled parameters to tell the RGS Sender to match the resolution and display layout of the client device. The configuration file also sets the IsMutable value for these parameters to 0, so the user’s local RGS client does not override the protocol plan.

When the user establishes a connection, the RGS Sender attempts to match the resolution and display layout. If the Sender cannot perform the match, the Sender reverts to its previous resolution.

⚠️ The user receives no warning that the Sender failed to match the resolution.

**Activating HP Velocity and Advanced Video Compression Features**

The HP Velocity and Advanced Video Compression features improve RGS performance over WAN connections. You can utilize these new features when establishing RGS connections using Leostream Connect when using RGS. The HP Velocity feature does not require additional configuration. To configure advanced video compression, include the following parameters in the HP RGS configuration file in your protocol plan.
• Rgreceiver.ImageCodec.IsH264Enable: Set to 1 to enable advanced video compression.

• Rgreceiver.ImageCodec.IsCPUEncode: Set to 1 to cause the RGS Sender to use CPU encoding for h.264. If this parameter is set to zero, the RGS Sender uses the GPU for encoding, if available.

The advanced video compression and HP Velocity functionality available in RGS require activation the first time the RGS Receiver connects to the RGS Sender. When connecting natively from the RGS Receiver to RGS Sender, activation dialogs open, indicating if the activation succeeded or failed. Leostream Connect suppresses the activation dialogs, however the activation continues to take place.

If you configured a proxy within RGS to perform the activation, include the following three parameters in the RGS configuration file in your protocol plan.

• Rgreceiver.Network.ProxyEnabled: Set to 1 to enable the proxy, if required, in the environment

• Rgreceiver.Network.ProxyPort: Specify the proxy port

• Rgreceiver.Network.ProxyAddress: Specify the proxy hostname or IP address

RGS uses the system proxy settings, but only when manual proxy configuration is enabled. RGS does not support the use of use PAC, WPAD, or proxy authentication. If there is no internet access and no proxy possible, RGS fails to activate and disables the HP Velocity and Advanced Video Compression features.

If the activation fails, use the following Rgreceiver parameters to configure the resultant behavior.

• Rgreceiver.Activation.AutomationMode: Specifies the path to take if the activation fails, either:
  
  o 0 – Continue without activation: in this mode, the RGS Receiver silently disables features requiring activation (HP Velocity and Advanced Video Compression) for the current session and continues with the connection. The next RGS connection triggers activation again.

  o 1 – Retry the activation: in this mode, the RGS Receiver retries activation before falling back. The number of retries is controlled by the Rgreceiver.Activation.RetryAttempts parameter.

  o 2 – (default) Do not activate: in this mode, the RGS Receiver disables the features that require activation. On the next connection if the user has not re-enabled those features, no activation attempt will occur.

• Rgreceiver.Activation.RetryAttempts: (default = 5) The number of reactivation attempts before disabling features that require activation
Setting User Configurable HP RGS Parameters

The configuration file in the HP RGS section of the protocol plan defines the characteristics of the user’s RGS session. These settings may override any parameter settings made on the user’s RGS Receiver.

In some cases, you may want the user to customize certain RGS connection parameters, including:

- Borders
- Resolution, including resolution and display layout
- Image quality
- Setup mode sequence

To allow users to set values for these parameters, select the **Allow users to modify configuration file parameters** option in the protocol plan.

To configure which parameters the user is allowed to modify:

1. Select the checkbox before each parameter that the user can customize.

2. After selecting the parameters the user can control, modify the text in the protocol plan’s **Configuration file** field to use pre-defined dynamic tags, described in the following table, which the Connection Broker replaces at connection time with the values specified by the user.

⚠️ If you do not place the dynamic tags in the **Configuration file**, the user-specified settings will not be applied. Consult the HP RGS user’s guide available at [http://www.hp.com/go/rgs](http://www.hp.com/go/rgs) for more information on the proper syntax for configuring RGeceiver parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>RGeceiver Parameter in the Protocol Plan</th>
<th>Leostream Dynamic Tag</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup mode</td>
<td>.Hotkeys.SetupModeSequence</td>
<td>{SETUP_MODE_SEQUENCE}</td>
</tr>
<tr>
<td>Show borders</td>
<td>.IsBordersEnabled</td>
<td>{BORDERS}</td>
</tr>
<tr>
<td>Match resolution</td>
<td>.IsMatchReceiverResolutionEnabled</td>
<td>{MATCH_RESOLUTION}</td>
</tr>
<tr>
<td>Match layout</td>
<td>.IsMatchReceiverPhysicalDisplaysEnabled</td>
<td>{MATCH_DISPLAYS}</td>
</tr>
<tr>
<td>Resolution</td>
<td>.Session.{SESSION}.VirtualDisplay.IsPreferredResolutionEnabled .Session.{SESSION}.VirtualDisplay.PreferredResolutionHeight .Session.{SESSION}.VirtualDisplay.PreferredResolutionWidth</td>
<td>{RESOLUTION_ENABLED} {RESOLUTION_HEIGHT} {RESOLUTION_WIDTH}</td>
</tr>
<tr>
<td>Image quality</td>
<td>.ImageCodec.Quality</td>
<td>{IMAGE_QUALITY}</td>
</tr>
</tbody>
</table>

For example, to allow the user to configure borders, image quality, and the setup mode sequence, you must add the following lines to your configuration file.
The Connection Broker does not replace the \{SESSION\} dynamic tag. Instead, Leostream Connect automatically adjusts the value for the \{SESSION\} dynamic tag when the user connects to desktops using HP RGS.

3. From the Default value drop-down menus, indicate the value to use if the user has not customized the parameter.

4. If you select Custom for the default value for resolution, enter the custom value into the Default custom value edit field. Enter the value as height\times width where \text{height} and \text{width} are in pixels and there is no space between the numbers and the \text{x}.

5. The drop-down menus in the end-user dialog display the values shown in the Default value drop-down menu on the Administrator interface. You can choose to show user-friendly descriptions of these items by defining display values. To define display values:
   a. Click the Edit link in the Display value column
   b. In the Edit Display Values form that opens, enter user-friendly names into the Display value edit field for each possible internal value.
   c. Click Save on the Edit Display Values form. The new display values are shown in the Default value drop-down menu, as they will be displayed to users.

If the user never opts to customize values for the configurable protocol plan parameters, their connections open using the default values specified in the protocol plan. If the user does specify a customized value for a parameter, the scope of that parameter is determined by the user’s policy. See User Configurable Protocol Plan Parameters for information on how to define the scope of user-configurable parameters, as well as for instructions on using the end-user interfaces to define parameter values.

Single Sign-On with HP RGS

The Leostream Agent is not responsible for signing the user onto the remote desktop. Instead, to achieve single sign-on with HP RGS:

- When installing the RGS Sender on a Windows desktop, ensure that the HP RGS Single Sign-on option is configured.
- For Linux remote desktop, ensure that the RGS Sender interacts with a PAM module that requests only the username and password, in that order, for authentication.
USB Passthrough with HP RGS

To connect USB devices to the remote Windows desktop, use either the HP USB redirector or Leostream Connect USB management. For predictable behavior, do not use these two features, simultaneously. If you use Leostream Connect USB management, you cannot use the Assign to active desktop USB option in the When Device is Plugged In section (see USB Device Management).

When using the Java version of Leostream Connect and the HP USB redirector, you can use the Leostream Connect sidebar to select which active remote session has access to all USB devices.

To turn on the sidebar for USB access:

1. Enable the sidebar by adding the following line to the lc.conf file on the client device.

   ```
   sidebar_enabled = true
   ```

2. In the protocol plan assigned to users that connect to desktops using HP RGS, add the following line to the Configuration file field for HP RGS.

   ```
   Rgreceiver.Usb.ActiveSession={USB_SESSION}
   ```

   The Windows version of Leostream Connect does not support the `{USB_SESSION}` tag.

When a user logs in through Leostream Connect, by default, the first desktop they connect to using HP RGS has access to all USB devices. The sidebar menu for this desktop, shown in the following figure, displays a USB On menu item.

When you attach a USB device to your client device, the USB device appears in the remote desktop that indicates USB On. You can switch all USB devices to another desktop by selecting the Turn USB On menu associated with that desktop, as shown in the following figure.

You must be connected to the desktop using RGS before you can connect USB devices. Leostream Connect prompts you to confirm that all USB devices should be switched to the new desktop. Click Yes in the confirmation dialog, shown in the following figure to move USB devices to the new desktop. Click No to keep the USB devices attached to the current desktop.
HP RGS simultaneously allows access to USB devices from a single desktop.

If you disconnect from the RGS session that has access to USB devices, Leostream Connect automatically switches all USB devices to the next active RGS session.

**Session Shadowing and Collaboration**

The Connection Broker allows users that connect to desktops using the HP RGS protocol to collaborate by inviting a second *shadow* user to connect to their session. You must configure the HP RGS Sender to accept collaborators before using Leostream to send invitations.

For information on enabling collaboration in the Connection Broker and sending invitations, see [Session Shadowing and Collaboration](#).

**Using the HP RGS Receiver for macOS**

You can install the Java version of Leostream Connect on an Apple macOS device and use the HP RGS Receiver for Mac to establish RGS connections. See the [Leostream Installation Guide](#) for information on installing Leostream Connect on a Mac.

Leostream Connect does not automatically discover the location of the HP RGS Receiver app. Specify the full path to the HP RGS Receiver executable in the Viewers tab of the Leostream Connect Options dialog. Do not enter the path to the .app directory, for example:

```
/Applications/HP RGS Receiver.app/Contents/MacOS/HP RGS Receiver
```

**Using HP SAM Clients**

To connect to your Leostream Connection Broker from an HP SAM client device, enter your Connection Broker IP address or DNS name into the SAM Server field on the HP SAM Client dialog, shown in the following figure.

To configure the remote viewer session that is launched from the HP SAM client, edit the protocol plan.
associated with the desktops that are launched from the SAM client and scroll down to the HP SAM Configuration section.

Enter the Configuration file in XML-format. The parameters in this file map to individual controls on the SAM client’s Options user interface.

Understanding the Configuration File

The configuration file allows you to customize the end user’s experience when they log into the Connection Broker using an HP SAM client. The parameters set in this file are similar to those set in the Global Policy in the HP Session Allocation Manager Web interface.

The parameters within the <Policy> section pertain to RGS and RDP connections. The parameters inside the <DynamicPolicy> tags pertain specifically to RGS.

Use the following format to specify parameters in the <Policy> section:

```xml
<Span V="0" O="1" />
```

Where the string equated to V is the value to assign to that parameter. The string equated to O indicates if the end user can override the policy setting using the options on the SAM client.

- O="1" indicates that the setting on the SAM client overrides the policy setting.
- O="0" indicates that the SAM client cannot override the policy setting.

⚠️ By default, the HP SAM configuration file allows the user to override the policy settings. Ensure that you switch the override values to zero to hard-code the behavior.

Use the following format to specify RGS Receiver parameters in the <DynamicPolicy> tags.

```xml
```

Where value can be any of the following:

- No value: Leave the tag empty to not specify this parameter
- 1: To enable this setting
- 0: To disable this setting

By default, the RGS Receiver parameters can be modified by the client device. To hard-code the policy behavior, add an isMutable tag for each RGS Receiver property to hard-code. For example, to ensure the end user cannot turn off the RGS feature to match the client device resolutions, add the following line to the configuration file.

```xml
```
**Configuring HP SAM for Multi-Monitor Support**

The default HP SAM configuration file does not provide multi-monitor support. To provide multi-monitor support:

1. **Modify the Span and Display settings in the <Policy> section, as follows:**

   ```xml
   <Span "V=1" O="0" />
   <Display FS="1" X="-1" Y="-1" Depth="-1" Stretch="0" O="0" />
   ```

2. **Add the following lines to the <DynamicPolicy> section.**

   ```xml
   ```

3. **If your RGS session opens with boarders, ensure the IsBordersEnabled parameter is set to zero:**

   ```xml
   ```

**Policy Options for HP SAM**

For users connecting to and from Windows machines, you can control the time zone of the remote desktop, using the **Adjust time zone on the destination to match client when user logs in** policy option. If the user’s policy selects this option, the Connection Broker changes the time zone of the desktop the user logs into to the same time zone as on the client device they log in from.

**Microsoft® RDP and RemoteFX**

The **RDP and RemoteFX** section of the protocol plan allows you to enter command line parameters and/or a configuration file to use when launching a Microsoft remote desktop connection. The Connection Broker uses the standard Microsoft RDP configuration file format for RDP sessions controlled by Leostream Connect. You can verify the configuration file you enter in the **Configuration file** edit field of the protocol plan using a standard Microsoft RDP client.

**Options for Encoding Desktop Login Credentials into RDP Configuration Files**

RDP requires an encrypted password in order to perform single sign-on. Typically, the Configuration file for RDP contains the following line:

```plaintext
password 51:b:{RDP_PASSWORD}
```

Using the `{RDP_PASSWORD}` dynamic tag in the protocol plan encodes the user’s desktop login credentials into the RDP configuration file. The Connection Broker replaces the `{RDP_PASSWORD}`
dynamic tag with the user’s password encrypted for RDP connections before passing the RDP configuration file to the client.

Do not use the `password 51:b` parameter in the configuration file. The Web client cannot encrypt the user’s password.

If the user’s desktop requires a different password than what the user provided to Leostream Connect, you can use the `{STANDARD_RDP_PASSWORD:password}` dynamic tag to pass the desktop password down to the Leostream Connect client in order to enable single sign-on. In your configuration file, replace `password` with the password to log into the desktop. Leostream Connect then encrypts the password and places the encrypted password in the configuration file before launching the RDP connection.

### Launching RDP Connections from the Leostream Web client

When launching an RDP connection from the Leostream Web client, the Connection Broker downloads an RDP configuration file in the same way as any other server-initiated file download. If your Web browser blocks the download, modify the browser’s security settings to allow downloads from your Connection Broker. After the browser downloads the file, it prompts the user to open or save the file. Opening the file launches the RDP session, where the user must enter their password.

Single sign-on is not available when using the native RDP client from a Web browser.

Some Web browsers prompt users to download the RDP file used to launch the Connection to the desktop. To avoid this prompt, the first time the Connection Broker tries to download an RDP file, right-click on the download tab associated with that file and select **Always open this kind of file**. When the user subsequently launches additional desktops, the Web browser automatically launches the connection without prompting the user.

Ensure that you do not save credentials from the client device’s native RDP client. If a user previously saved credential by checking the **Allow me to save credentials** option on the Remote Desktop Connection login page, ensure that you delete the credential before trying to log into that remote desktop from the Leostream Web client.

### Microsoft RDP Viewer Command Line Parameters

The following is a list of some useful RDP command line parameters. For an online description all the RDP command line parameters, go to the following Microsoft Windows support page.


`/f`
Start the RDP connection in full-screen mode.

`/span`
Use this parameter to span across multiple monitors with the same height and width.
/w:<width>
Specify the width of the RDP connection windows.

/h:<height>
Specify the height of the RDP connection window.

Microsoft RDP Viewer Configuration File Variables

The following is a list of the RDP file parameters contained in the default configuration file for new protocol plans. Where Connection Broker dynamic tags are included in the parameter name, ensure that you include the dynamic tag when using that parameter in the configuration file contained in the protocol plan.

**use multimon:i:** (RDP 7, only)
Indicates if the remote session should span across all monitors attached to the client device. When using this option, the monitors do not have to have the same resolution and orientation. If using RDP 6, use **span monitors**, instead.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Use a single monitor</td>
</tr>
<tr>
<td>1</td>
<td>Use all monitors</td>
</tr>
</tbody>
</table>

**span monitors:i**
Indicates if the remote session should be spanned across multiple monitors.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spanning is off</td>
</tr>
<tr>
<td>1</td>
<td>Spanning is on</td>
</tr>
</tbody>
</table>

**screen mode id:i**
Determines if the remote session is opened in a window or in full screen.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Open in full screen</td>
</tr>
<tr>
<td>1</td>
<td>Open in a window</td>
</tr>
</tbody>
</table>

**desktopwidth:i**
Corresponds to the desktop width (in pixels) on the **Display** tab in Remote Desktop Connection Options dialog. 

For Microsoft ActiveX® remote viewers, the variables are:

MsRdpClient.DesktopWidth = screen.width
MsRdpClient.width = screen.width
**desktopheight:**
Corresponds to the desktop height (in pixels) on the **Display** tab in Remote Desktop Connection **Options** dialog.

For ActiveX remote viewers, the variables are:

```java
MsRdpClient.DesktopHeight = screen.height
MsRdpClient.height = screen.height
```

**connection type:**
Corresponds to the selection in the **Choose your connection speed to optimize performance** drop-down on the **Experience** tab in Remote Desktop Connection **Options** dialog. To invoke RemoteFX, set this value to 6, and set the **session bpp** parameter to 32.

**session bpp:**
Corresponds to the color depth you select in the **Colors** drop-down on the **Display** tab in Remote Desktop Connection **Options** dialog. To invoke RemoteFX, set this value to 32, and set the **connection type** parameter to 6.

**winposstr:**
Corresponds to the window position on the **Display** tab in Remote Desktop Connection **Options** dialog.

On desktop computers, this setting determines the Remote Desktop Connection dialog box position on the screen. The six numbers represent a string form of the **WINDOWPOS** structure. For more information about the **WINDOWPOS** function, visit the following Microsoft Web page:


**auto connect:**
This setting is not used by desktop computers or by Windows CE-based clients.

**Full Screen Title**
For ActiveX remote clients, the variable is:

```java
MsRdpClient.FullScreenTitle =
```

**full address:**
Determines the IP address of desktop. The setting corresponds to the entry in the **Computer** field on the **General** tab of Remote Desktop Connection **Options** dialog. The Connection Broker can dynamically set this property using the **{IP}** or the **{Windows_Name}** dynamic tag.

For ActiveX remote viewers, the variable is:

```java
MsRdpClient.server = "{IP}"
```

**password:**
Controls password settings.

RDP requires an encrypted password to perform single sign-on. The Connection Broker passes as unencrypted password to the client device; the client device is then responsible for password encryption.
Using the `{RDP_PASSWORD}` tag, the Windows version of Leostream Connection encrypts the password and places the encrypted version into the configuration file, resulting in single sign-on to the desktop.

⚠️ The Java version of Leostream Connect and the Connection Broker Web client cannot encrypt the RDP password. Use the plain password option when using RDP to connect to a desktop from the Java version of Leostream Connect or the Web client.

`password:s: {PLAIN_PASSWORD}`

Controls password settings.

In this case, the Connection Broker sends a plain-text password to the client device. Use this option if launching Microsoft RDP connections from the Java version of Leostream Connect, the Leostream Web client, or thin clients from vendors such as HP that write to the Leostream API.

For ActiveX remote viewers, the configuration parameter is:

```
MsRdpClient.AdvancedSettings.ClearTextPassword="{PLAIN_PASSWORD}"
```

`compression:i`

Determines if data is compressed when it is transmitted to the client computer, according to the following values:

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Compression is off</td>
</tr>
<tr>
<td>1</td>
<td>Compression is on</td>
</tr>
</tbody>
</table>

`keyboardhook:i`

Determines where Windows key combinations are applied. This setting corresponds to the selection in the Keyboard section on the Local Resources tab of Remote Desktop Connection Options dialog, according to the following values:

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>On the local computer</td>
</tr>
<tr>
<td>1</td>
<td>On the remote computer</td>
</tr>
<tr>
<td>2</td>
<td>In full-screen mode only</td>
</tr>
</tbody>
</table>

`audiomode:i`

Determines where sounds are played. This setting corresponds to the selection in the Remote computer sound section on the Local Resources tab of Remote Desktop Connection Options dialog, according to the following values:

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Bring to this computer</td>
</tr>
<tr>
<td>1</td>
<td>Leave at remote computer</td>
</tr>
<tr>
<td>2</td>
<td>Do not play</td>
</tr>
</tbody>
</table>
**redirectclipboard:i:1**
Determines if the clipboard is enabled in the remote session. This setting corresponds to the selection of the **Clipboard** option in the **Local devices and resources** section on the **Local Resources** tab of Remote Desktop Connection **Options** dialog, according to the following rules.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Clipboard is not enabled</td>
</tr>
<tr>
<td>1</td>
<td>Clipboard is enabled</td>
</tr>
</tbody>
</table>

**redirectdrives:i:0**
Determines if disk drives are automatically connected when you log on to the remote desktop. This setting corresponds to the selection of the **Drives** option in the **More Local devices and resources** dialog, accessed via the **More** button in the **Local Resources** tab of Remote Desktop Connection **Options** dialog. The following values apply.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Drives are not automatically reconnected</td>
</tr>
<tr>
<td>1</td>
<td>All drives are automatically reconnected</td>
</tr>
</tbody>
</table>

For ActiveX remote viewers, the variable is:

```plaintext
MsRdpClient.AdvancedSettings2.RedirectDrives = FALSE
```

**drivestoredirect:s**
Determines which drives are automatically connected when you log on to the remote desktop. Use `drivestoredirect:s:*` to redirect all existing drives and any subsequently connected drives. To redirect a specific drive, enter the drive name followed by a colon, for example, `drivestoredirect:s:C:`. To redirect multiple drives, use a semi-colon to separate the drive names. Use the **DynamicDrives** tag to redirect drives that are connected to the client after the remote session is established. For example, the following parameter redirects the C drive and dynamic drives:

```
drivestoredirect:s:C:;DynamicDrives
```

The Windows version of Leostream Connect supports the following two dynamic tags when connecting using RDP 6. These two dynamic tags are supported for RDP 7 only when the RDP 7 client is installed on a Windows XP operating system and the drive is referenced as the volume label followed by the drive letter. Leostream Connect cannot redirect drives using RDP 7 if the drives are referenced by the drive label followed by the drive letter, or by a combination of drive label, drive letter, and volume label.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>{DRIVE:DVD}</td>
<td>All DVD drives are automatically connected. No other drives are connected.</td>
</tr>
<tr>
<td>{DRIVE:CD}</td>
<td>All CD drives are automatically connected. No other drives are connected.</td>
</tr>
</tbody>
</table>
**devicestoredirect:**
Determines which supported Plug and Play devices on the client computer are automatically redirected when you log on to the remote desktop. Use `devicestoredirect:s:*` to redirect all supported Plug and Play devices. To redirect any supported Plug and Play devices that are connected later, use the `DynamicDevices` tag. For example, `devicestoredirect:s:DynamicDevices`.

**redirectposdevices:**
Determines whether media players based on the Media Transfer Protocol (MTP) and digital cameras based on the Picture Transfer Protocol (PTP) are redirected. This setting corresponds to the **Supported Plug and Play devices** option in the **More Local devices and resources** dialog, accessed via the **More** button in the **Local Resources** tab of Remote Desktop Connection **Options** dialog. The following values apply.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Microsoft Point of Service for .NET (POS for .NET) device redirection is disabled</td>
</tr>
<tr>
<td>1</td>
<td>Microsoft Point of Service for .NET (POS for .NET) device redirection is enabled</td>
</tr>
</tbody>
</table>

**redirectprinters:**
Determines whether printers are automatically connected when you log on to the remote computer. This setting corresponds to the selection in the **Printers** check box in the **Local devices and resources** section on the **Local Resources** tab of Remote Desktop Connection **Options** dialog. The following values apply.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Printers are not automatically reconnected</td>
</tr>
<tr>
<td>1</td>
<td>Printers are automatically reconnected</td>
</tr>
</tbody>
</table>

For ActiveX remote viewers, the variable is:

```
MsRdpClient.AdvancedSettings2.RedirectPrinters = TRUE
```

**redirectcomports:**
Determines if COM ports are automatically connected when you log on to the remote computer. This setting corresponds to the **Serial Ports** option in the **More Local devices and resources** dialog, accessed via the **More** button in the **Local Resources** tab of Remote Desktop Connection **Options** dialog. The following values apply.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>COM ports are not automatically reconnected</td>
</tr>
<tr>
<td>1</td>
<td>COM ports are automatically reconnected</td>
</tr>
</tbody>
</table>
redirectsmartcards:
Determines if smart cards are automatically connected when you log on to the remote computer. This setting corresponds to the Smart cards box in the More Local devices and resources dialog, accessed via the More button on the Local Resources tab of Remote Desktop Connection Options dialog. The following values apply.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Smart cards are not automatically reconnected</td>
</tr>
<tr>
<td>1</td>
<td>Smart cards are automatically reconnected</td>
</tr>
</tbody>
</table>

For ActiveX remote viewers, the variable is:

```csharp
MsRdpClient.AdvancedSettings2.RedirectSmartCards = FALSE
```

displayconnectionbar:
Determines whether the connection bar is displayed when you log on to the remote computer in full-screen mode. This setting corresponds to the selection in the Display the connection bar when in full screen mode option on the Display tab of Remote Desktop Connection Options dialog. The following values apply.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Connection bar does not appear</td>
</tr>
<tr>
<td>1</td>
<td>Connection bar appear</td>
</tr>
</tbody>
</table>

For ActiveX remote viewers, the variable is:

```csharp
MsRdpClient.AdvancedSettings3.ConnectionBarShowMinimizeButton = FALSE
```

username:
Determines the user account used to log into the desktop. This setting corresponds to the entry in the User name edit field on the General tab of Remote Desktop Connection Options dialog. The Connection Broker can dynamically set this property using the {USER} field.

For ActiveX remote viewers, the variable is:

```csharp
MsRdpClient.server = "{IP}" 
```

domain:
Determines the domain used to authenticate the user. This setting corresponds to the entry in the Domain edit field on the General tab of Remote Desktop Connection Options dialog. The Connection Broker can dynamically set this setting using the {DOMAIN} field.

For ActiveX remote viewers, the variable is:

```csharp
MsRdpClient.Domain = "{DOMAIN}" 
```
**alternate shell:**
Determined if a program is started automatically when you connect with RDP. The setting corresponds to the entry in the Program path and file name edit field on the Programs tab of Remote Desktop Connection Options dialog.

**shell working directory:**
Indicates the starting folder for the application that is automatically started when you connect with RDP. The setting corresponds to the entry in the Start in the following folder edit field on the Programs tab of Remote Desktop Connection Options dialog.

**disable wallpaper:**
Determined if the desktop background appears when you log on to the remote computer. This setting corresponds to the selection in the Desktop background check box on the Experience tab of Remote Desktop Connection Options dialog, according to the following values.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Wallpaper appears</td>
</tr>
<tr>
<td>1</td>
<td>Wallpaper does not appear</td>
</tr>
</tbody>
</table>

**disable full window drag:**
Determined if folder contents appear when you drag the folder to a new location. This setting corresponds to the selection in the Show contents of window while dragging check box on the Experience tab of Remote Desktop Connection Options dialog, according to the following values.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Folder contents appear while dragging</td>
</tr>
<tr>
<td>1</td>
<td>Folder contents do not appear while dragging</td>
</tr>
</tbody>
</table>

**disable menu anims:**
Determined how menus and windows appear when you log on to the remote computer. This setting corresponds to the selection in the Menu and window animation check box on the Experience tab of Remote Desktop Connection Options dialog, according to the following values.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Menu and window animations are permitted</td>
</tr>
<tr>
<td>1</td>
<td>Menu and window animations are not permitted</td>
</tr>
</tbody>
</table>

**disable themes:**
Determined if themes are permitted when you log on to the remote computer. This setting corresponds to the selection in the Themes check box on the Experience tab of Remote Desktop Connection Options dialog, according to the following values.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Themes are permitted</td>
</tr>
<tr>
<td>1</td>
<td>Themes are not permitted</td>
</tr>
</tbody>
</table>
Working with Display Protocols

bitmapcachepersistenable:i
Determines if bitmaps are cached on the local computer. This setting corresponds to the selection in the Bitmap caching check box on the Experience tab of Remote Desktop Connection Options dialog, according to the following values.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Caching is not enabled</td>
</tr>
<tr>
<td>1</td>
<td>Caching is enabled</td>
</tr>
</tbody>
</table>

Connecting to RemoteApp Servers

To offer the user RemoteApp sessions, first create a Remote Desktop Services/Multi-User center for the Windows Server that publishes the applications (see “Remote Desktop Services / Multi-User Centers” in the Leostream Connection Broker Administrator’s Guide). When creating the center, indicate the maximum number of users that can simultaneously log into the server.

Next, group the RemoteApp sessions created by the center into pools on the > Configuration > Pools page. You can publish all the applications from a single pool or separate the sessions into individual pools for each application. If you plan to offer multiple applications to a particular user, separate the sessions into multiple pools, by application.

Finally, if you are connecting directly to the RemoteApp server, use the following parameters in the RDP Configuration File to launch a particular application on the RemoteApp server. Each protocol plan publishes a single application. Create as many protocol plans as needed, based on the number of applications published on the server.

If you will connect to the RemoteApp server through the Leostream Gateway, see the Leostream Gateway Guide for information on how to build an appropriate protocol plan.

remoteapplicationmode:i
(Required) Determines whether a RemoteApp should be launched when connecting to the remote computer.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Use a normal session and do not start a RemoteApp.</td>
</tr>
<tr>
<td>1</td>
<td>Connect and launch a RemoteApp.</td>
</tr>
</tbody>
</table>

remoteapplicationname:s
(Required) Specifies the name of the RemoteApp in the Remote Desktop interface while starting the RemoteApp.

remoteapplicationprogram:s
(Required) Specifies the alias or executable name of the RemoteApp.
**remoteapplicationcmdline:**
Optional command line parameters for the RemoteApp.

**remoteapplicationfile:**
Specifies a file to be opened on the remote computer by the RemoteApp.

When building policies to offer RemoteApp sessions, associate each protocol plan with the appropriate pool of sessions. A Leostream policy can offer sessions from multiple pools, allowing you to publish multiple applications in one policy. However, a particular pool can only be referenced once, per policy.

By default, RemoteApp servers keep the user logged into the session even after the disconnect from their application. If you do not want to modify the group policy on your RemoteApp server, you can use Leostream release plans to log the users out of their disconnected sessions. See the “Release Plans” section in the Leostream Connection Broker Administrator’s Guide for more information.

### Integrating with a Microsoft Remote Desktop Gateway

Use the following parameters in the RDP Configuration File to support connections to desktops via a Microsoft Remote Desktop Gateway.

**gatewaycredentialssource:**
Specifies the credentials that should be used to validate the connection with the RD Gateway.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ask for password (NTLM)</td>
</tr>
<tr>
<td>1</td>
<td>Use smart card</td>
</tr>
<tr>
<td>4</td>
<td>Allow user to select later</td>
</tr>
</tbody>
</table>

**gatewayhostname:**
Specifies the hostname of the RD Gateway.

**gatewayprofileusagemethod:**
Determines the RD Gateway authentication method to be used.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Use the default profile mode, as specified by the administrator.</td>
</tr>
<tr>
<td>1</td>
<td>Use explicit settings.</td>
</tr>
</tbody>
</table>
Working with Display Protocols

`gatewayusagemethod:i`
Specifies if and how to use a Remote Desktop Gateway (RD Gateway) server.

<table>
<thead>
<tr>
<th>Value</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Do not use an RD Gateway server</td>
</tr>
<tr>
<td>1</td>
<td>Always use an RD Gateway, even for local connections.</td>
</tr>
<tr>
<td>2</td>
<td>Use the RD Gateway if a direct connection cannot be made to the remote computer (i.e. bypass for local addresses)</td>
</tr>
<tr>
<td>3</td>
<td>Use the default RD Gateway settings</td>
</tr>
<tr>
<td>4</td>
<td>Do not use an RD Gateway server (default)</td>
</tr>
</tbody>
</table>

**Mechdyne TGX**

Mechdyne TGX delivers high resolution without sacrificing image quality or impacting performance. End users can launch TGX connections from either the Leostream Connect client or using the Leostream Web client.

⚠️ Connection Broker 9.0 supports TGX version 1.10. To use older versions of TGX, please remain on Connection Broker 8.2.

**Launching Mechdyne TGX Connections**

You must install the Mechdyne TGX client on any device that will connect to remote desktops using TGX. Leostream launches the TGX client using the TGX-file configured by the user’s Leostream protocol plan. To configure a protocol plan to use TGX:

1. Scroll down to the **Mechdyne TGX** section of the protocol plan in the **Leostream Connect and Thin Clients Writing to Leostream API** or **Web Browser** sections.

2. Change the **Priority** to 1.

3. If any other protocol in the associated section of the protocol plan has a priority of 1, modify that protocol’s priority to a lower number or to **Do not use**.

4. Use the **Configuration file** field associated with TGX to customize the remote session. The configuration file specifies parameters in the `.tgx` file used to launch the Mechdyne client.

The default configuration file uses the `{CREDENTIALS_MECHDYNE}` dynamic tag. The Connection Broker uses a proprietary Mechdyne algorithm to encrypt the user’s credentials to pass to the TGX Sender and provide single sign-on.
Setting User-Configurable TGX Parameters

The configuration file in the Mechdyne TGX section of the protocol plan defines the characteristics of the user’s TGX session. The values in the protocol plan configuration file override any parameter settings made on the user’s TGX client.

You can allow the user to configure the following TGX connection parameters:

- Image quality
- Client resolution
- Local mouse redirection
- Update rate

To allow users to set values for these parameters, select the **Allow users to modify configuration file parameters** option in the protocol plan. The form expands to include the additional fields shown in the following figure.

To configure which parameters the user is allowed to modify:

1. Select the checkbox before each parameter that the user can customize.

2. Modify the text in the protocol plan’s **Configuration file** field to use the dynamic tags associated with the selected parameters, as described in the following table. The Connection Broker replaces the dynamic tags at connection time with the values specified by the user.

⚠️ If you do not place the dynamic tags in the **Configuration file**, the user-specified settings will not be applied.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>TGX Parameter in the Protocol Plan</th>
<th>Leostream Dynamic Tag</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image quality</td>
<td>imageQuality</td>
<td>{IMAGE_QUALITY}</td>
</tr>
<tr>
<td>Match my desktop</td>
<td>matchDesktop</td>
<td>{MATCH_DESKTOP}</td>
</tr>
<tr>
<td>Mouse redirection</td>
<td>showLocalMouse</td>
<td>{MOUSE_REDIRECTION}</td>
</tr>
<tr>
<td>Update rate</td>
<td>updateRate</td>
<td>{UPDATE_RATE}</td>
</tr>
</tbody>
</table>

For example, if the user can configure all four parameters, the configuration file looks as follows.

```text
removeFile=false
hostname={IP}
domain={DOMAIN}
credentials={CREDENTIALS_MECHDYNE}
matchDesktop={MATCH_DESKTOP}
showLocalMouse={MOUSE_REDIRECTION}
imageQuality={IMAGE_QUALITY}
updateRate={UPDATE_RATE}
```

3. Use the **Default value** drop-down menus to indicate the default value to use if the user does not customize the parameter.

4. The drop-down menus in the end-user dialog display the values shown in the **Default value** drop-down menu on the Administrator interface. You can choose to show user-friendly descriptions of these items by defining display values. To define display values:
   
   a. Click the **Edit** link in the **Display value** column.
   
   b. In the **Edit Display Values** form that opens, enter user-friendly names into the **Display value** edit field for each possible internal value.
   
   c. Click **Save** on the **Edit Display Values** form. The new display values are shown in the **Default value** drop-down menu, as they will be displayed to users.

If the user does not customize values for the configurable protocol plan parameters, their connections open using the default values specified in the protocol plan. If the user does specify a customized value for a parameter, the scope of that parameter is determined by the user’s policy. See **User Configurable Protocol Plan Parameters** for information on how to define the scope of user-configurable parameters, as well as for instructions on using the end-user interfaces to define parameter values.
NICE Desktop Cloud Visualization (DCV)

**NICE Desktop Cloud Visualization** (DCV) is an advanced technology that enables users to remote access 2-D and 3-D interactive applications over a standard network. End users can launch DCV connections from either the Leostream Connect client or using the Leostream Web client.

Launching NICE DCV Connections

You must install the NICE DCV end station on any device that will connect to remote desktops using DCV. Leostream launches the DCV client using the VNC-file configured by the user’s Leostream protocol plan. To configure a protocol plan to use NICE DCV:

1. Scroll down to the **NICE DCV and VNC** section of the protocol plan in the **Leostream Connect and Thin Clients Writing to Leostream API** or **Web Browser** sections.

2. Change the **Priority** to 1.

3. If any other protocol in the associated section of the protocol plan has a priority of 1, modify that protocol’s priority to a lower number or to **Do not use**.

4. Use the **Configuration file** field associated with DCV to customize the remote session. The configuration file specifies parameters in the `.dcv` file used to launch the DCV client.

When using Leostream Connect, the command line parameters do not apply.

When a user requests a DCV connection to a desktop, the Connection Broker contacts the Leostream Agent running on the desktop to start the underlying VNC session for the user. After starting the session, the Leostream Agent returns the port associated with the session to the Connect Broker.

The **Connection Broker replaces the `{PORT}` dynamic tag in the protocol plan’s configuration file with the value returned by the Agent, before sending the VNC-file to the client device.**
NoMachine and FreeNX

The NoMachine section of the protocol plan allows you to specify a default configuration file for the NoMachine connection, as well as indicate any protocol plan parameters that are user configurable. End users can launch NoMachine connections from either the Leostream Connect client or using the Leostream Web client. To configure a protocol plan to use NoMachine:

1. Scroll down to the NoMachine NX section of the protocol plan in the Leostream Connect and Thin Clients Writing to Leostream API or Web Browser sections.

2. Change the Priority to 1.

3. If any other protocol in the associated section of the protocol plan has a priority of 1, modify that protocol's priority to a lower number or to Do not use.

4. Use the Configuration file field associated with NX to customize the remote session. The configuration file specifies parameters in the .nxs file used to launch the NX client.

5. If users are able to override the value for certain NX parameters, select the Allow user to modify configuration file parameters option. See Setting User-Configurable NoMachine Parameters for a list of configurable parameters and instructions on setting up this feature.

When creating a new protocol plan, the default configuration file supports NoMachine version 4.0. To support other versions of NoMachine, launch the NoMachine client, configure any desired client settings, and click Save. You can then copy-and-paste the contents of the resulting .nxs file into the Configuration file field. On a Windows client, the .nxs file is located in the user's .nx/config directory. After copying the file contents into the protocol plan, ensure that you replace certain strings with the necessary dynamic tags, as described in NX Configuration File.

To configure NoMachine 4 to mimic NX 3.5 behavior, view this NoMachine FAQ.

https://www.nomachine.com/AR07K00681

Launching NoMachine Connections from the Web Client

To setup a protocol plan for the Web client:

1. Go to the > Configuration > Protocol Plans page.

2. Create a new protocol plan or edit an existing plan.

3. In the Create Protocol Plan or Edit Protocol Plan form, scroll down to the Web Browser section.

4. Use the Configuration file field to define NX parameters for the connection. This field applies to the native NX client and NX Web Companion.
NoMachine Configuration File

The configuration file for NoMachine is an XML representation of the fields in the NoMachine Client GUI. The following table describes the important option keys when integrating with Leostream.

<table>
<thead>
<tr>
<th>Option Key</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>User</td>
<td>(Default = {USER}) The user’s login name. The Connection Broker replaces the dynamic tag {USER} with the name the user entered when logging in to the Connection Broker.</td>
</tr>
<tr>
<td>Auth</td>
<td>(Default = {SCRAMBLED_PASSWORD}) The user’s password. By default, the configuration file is configured to pass a scrambled password. If your NX server is configured to expect a plain password, replace the {SCRAMBLED_PASSWORD} dynamic tag in the Configuration file field with the {PLAIN_PASSWORD} dynamic tag. When the {SCRAMBLED_PASSWORD} is used, the Connection Broker uses the NoMachine method for scrambling passwords. This scrambled password is then passed in the configuration file.</td>
</tr>
<tr>
<td>Server host</td>
<td>(Default = {IP}) The Connection Broker replaces the dynamic tag {IP} with the hostname or IP address of the NoMachine server.</td>
</tr>
<tr>
<td>Public Key</td>
<td>Enter the key into the configuration file by placing an option key Public Key after the Login Method option, as follows.</td>
</tr>
</tbody>
</table>

```
...<option key="Login Method" value="nx" />
<option key="Public Key" value=""-----BEGIN DSA PRIVATE KEY-----
< Insert DSA Key here>
-----END DSA PRIVATE KEY-----
/>"
...
```

Session Shadowing and Collaboration

The Connection Broker allows users that connect to desktops using the NoMachine protocol to collaborate by inviting a second shadow user to connect to their session. You must configure the HP RGS Sender to accept collaborators before using Leostream to send invitations.

For information on enabling collaboration in the Connection Broker and sending invitations, see Session Shadowing and Collaboration.

Setting User-Configurable NoMachine Parameters

The configuration file in the NoMachine section of the protocol plan defines how the user’s NoMachine session is launched. These settings override any parameter settings made on the user’s NX software client.

In some cases, you may want the user to be able to customize certain NX connection parameters.
Currently, Leostream allows end-users to customize the following parameters.

- Connection type
- Desktop option
- Disable backingstore
- Resolution
- Span monitors
- Window manager

To allow users to set values for these parameters, select the **Allow users to modify NX configuration file parameters** option in the protocol plan. The form expands to include the additional fields shown in the following figure.

To configure which parameters the user is allowed to modify:

1. Select the checkbox before each parameter that the user can customize.

2. After selecting the parameters the user can control, modify the text in the protocol plan’s **Configuration file** field to use pre-defined dynamic tags, described in the following table, which the Connection Broker replaces at connection time with the values specified by the user.

⚠️ You cannot save the Protocol Plan form if you do not include all relevant dynamic tags.
### Working with Display Protocols

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Option Key</th>
<th>Original Text</th>
<th>Replace with</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connection type</td>
<td>Link speed</td>
<td>value=&quot;lan&quot;</td>
<td>value=&quot;{CONNECTION_TYPE}&quot;</td>
</tr>
<tr>
<td>Desktop option</td>
<td>Virtual desktop</td>
<td>value=&quot;false&quot;</td>
<td>value=&quot;{VIRTUAL_DESKTOP}&quot;</td>
</tr>
<tr>
<td>Disable backingstore</td>
<td>Disable backingstore</td>
<td>value=&quot;false&quot;</td>
<td>value=&quot;{BACKINGSTORE}&quot;</td>
</tr>
<tr>
<td>Resolution</td>
<td>Resolution</td>
<td>value=&quot;available&quot;</td>
<td>value=&quot;{RESOLUTION}&quot;</td>
</tr>
<tr>
<td></td>
<td>Resolution height</td>
<td>value=&quot;600&quot;</td>
<td>value=&quot;{RESOLUTION_HEIGHT}&quot;</td>
</tr>
<tr>
<td></td>
<td>Resolution width</td>
<td>value=&quot;800&quot;</td>
<td>value=&quot;{RESOLUTION_WIDTH}&quot;</td>
</tr>
<tr>
<td>Span Monitors</td>
<td>Spread over monitors</td>
<td>value=&quot;false&quot;</td>
<td>value=&quot;{SPAN_MONITORS}&quot;</td>
</tr>
<tr>
<td>Window manager</td>
<td>Desktop</td>
<td>value=&quot;kde&quot;</td>
<td>value=&quot;{WM_TYPE}&quot;</td>
</tr>
</tbody>
</table>

3. From the **Default value** drop-down menus, indicate the value to use before the user customizes the parameter.

4. If you select **Custom** for the default value for window manager or resolution, enter the custom value into the **Default custom value** edit field. For resolution, enter the value as `heightxwidth` where `height` and `width` are in pixels and there is no space between the numbers and the `x`.

5. The drop-down menus in the end-user dialog include all values shown in the **Default value** drop-down menu on the Administrator interface. You can choose to show user-friendly descriptions of these items by defining display values. To define display values:
   a. Click the **Edit** link in the **Display value** column
   b. In the **Edit Display Values** form that opens, enter user-friendly names into the **Display value** edit field for each possible internal value.
   c. Click **Save** on the **Edit Display Values** form. The new display values are shown in the **Default value** drop-down menu, as they will be displayed to users.

6. You must repeat steps 1 through 4 for the **Web browser** and **Leostream Connect** sections of the protocol plan, if users log in from both clients. The display values in step 5 can be specified only once. The Leostream Connect and Leostream Web clients use the same set of display values.

If the user never opts to customize values for the configurable protocol plan parameters, their connections open using the default values specified in the protocol plan. If the user does specify a customized value for a parameter, the scope of that parameter is determined by the user’s policy. See **User Configurable Protocol Plan Parameters** for information on how to define the scope of user-configurable parameters, as well as for instructions on using the end-user interfaces to define parameter values.
The Leostream Connection Broker can initiate PCoIP connections to the following types of remote desktops.

- Microsoft Windows and Linux workstations with installed PCoIP Remote Workstation cards. See the [Quick Start with Teradici PC-over-IP](#) for details on setting up the Connection Broker to manage PCoIP connections.

- Microsoft Windows and Linux virtual machines with installed Teradici Cloud Access Software.

- VMware virtual machines with an installed VMware Horizon View Agent Direct-Connection Plug-In.

Users can connect to virtual machines running the VMware Horizon View Agent Direct-Connection Plug-In using either the Leostream Web client, Leostream Connect, or a PCoIP Zero Client. In this case, a Leostream policy assigns the desktop to the user and, therefore, no VMware Horizon View Connection Server configuration is required.

### Using PCoIP Clients with Leostream

You can use any supported PCoIP software, mobile, or zero client to log into Leostream. The type of client you use and whether the client communicates with Leostream or the PCoIP Connection Manager determines what types of PCoIP resources can be connected. The following table describes the types of resources users can connect to from different PCoIP client.

<table>
<thead>
<tr>
<th>Client Type</th>
<th>Communicates with</th>
<th>Connects to Virtual Machines</th>
<th>Connects to Physical Machines</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCoIP Software</td>
<td>PCoIP Connection Manager</td>
<td>Running the PCoIP Standard or Graphics Cloud Access Software</td>
<td>N/A</td>
</tr>
<tr>
<td>(Mac, Windows, ChromeOS)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCoIP Mobile</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCoIP Zero Client</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCoIP Zero client</td>
<td>Leostream Connection Broker</td>
<td>Running the VMware Horizon View Direct Connection Plug-In</td>
<td>With an installed Remote Workstation Cards</td>
</tr>
</tbody>
</table>
## Enabling PCoIP Remote Workstation Card Support

Your Leostream license determines if you can manage connections to PCoIP Remote Workstation Cards. Contact sales@leostream.com if you need to enable PCoIP for your environment.

⚠ The PCoIP Remote Workstation Cards must run firmware version 4.9 to take advantage of all Leostream features.

After you enable the PCoIP feature, reboot your Connection Broker.

- The > Resources page contains a new PCoIP Host Devices section. The > Resources > PCoIP Host Devices page lists the PCoIP Remote Workstation cards registered with your Connection Broker.

- The > Setup > Centers page contains a new PCoIP Devices center. This center instructs the Connection Broker on how often to refresh the information associated with your PCoIP devices, as well as configures firmware updates and client bonding.

## Connecting to PCoIP Remote Workstation Cards

Users logging in from a PCoIP zero client can establish PCoIP connections to workstations with a PCoIP Remote Workstation card. Leostream supports connections to Windows and Linux operating systems. See the Quick Start with Teradici PC-over-IP for details on setting up the Connection Broker to manage hardware-based PCoIP connection.

To connect to workstations with an installed PCoIP Remote Workstation card using a PCoIP software client, the user must log into Leostream using Leostream Connect. Leostream Connect then launches the PCoIP software client to establish the PCoIP connection.

### Table: Client Type and Communication Details

<table>
<thead>
<tr>
<th>Client Type</th>
<th>Communicates with</th>
<th>Connects to Virtual Machines</th>
<th>Connects to Physical Machines</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leostream Connect</td>
<td>Leostream</td>
<td>• Running the VMware Horizon View Direct Connection Plug-In – Leostream Connect launches the VMware Horizon View Client to establish the connection</td>
<td>With an installed Remote Workstation Cards – Leostream Connect launches the PCoIP Software client to establish the connection</td>
</tr>
<tr>
<td></td>
<td>Connection Broker</td>
<td>• Running the Teradici Cloud Access Software – Leostream Connect launches the PCoIP software client to establish the connection</td>
<td></td>
</tr>
</tbody>
</table>
For this scenario, the user’s protocol plan must set the priority of the **Teradici PCoIP Soft Client** option to 1 and use the `--hard-host` command line parameter to specify the Remote Workstation Card to use for the connection, as shown in the following figure.

The Connection Broker replaces the `{PCOIP_HOST1}` dynamic tag with the IP address of the desktop’s Remote Workstation Card.

Ensure that the user’s policy selects the **Enable single sign-on to desktop console** option to provide single sign-on from the PCoIP software client to the remote desktop.

### Using the Teradici Cloud Access Software and Cloud Access Platform

For a complete description of how to use Leostream with the Teradici Cloud Access Software and Cloud Access Platform, please download the [Leostream Quick Start Guide for integrating with Teradici Cloud Access Software](#).

### PCoIP Connections to VMware Virtual Machines with a View Direct-Connection Plug-In

The Connection Broker can establish PCoIP connections to VMware virtual machines running the VMware Horizon View Direct-Connection Plug-in. The virtual machine must have an installed Leostream Agent.

⚠️ When installing the Leostream Agent, ensure that you do **not** select the **Credential Provider** task when performing the installation. The Leostream Agent credential provider may conflict with the Direct-Connection Plug-in.

Ensure that the PCoIP connection can be established from the VMware Horizon View Client to the virtual machine, before attempting to use with Leostream. You must configure the **View Agent Direct-Connection Users** on the virtual machine before Leostream can establish the PCoIP connection.
Users can connect to the desktop using the Leostream Web client, Leostream Connect, or a PCoIP zero
client.

Establishing Connections using Leostream Connect

When using Leostream Connect or the Leostream Web client, the user’s client device must have an installed
VMware Horizon View client. You can then use Leostream protocol plans to launch the VMware client and
establish a PCoIP connection to a Windows virtual machine running the VMware View Direct-Connection
Plugin.

To configure the protocol plan for software-based PCoIP connections:

1. Go to the > Configuration > Protocol Plans page.

2. Create a new protocol plan or edit an existing plan.

3. In the Leostream Connection and Thin Clients Writing to Leostream API section, select 1 from the
   Priority menu associated with VMware View.

4. Also in the Leostream Connection and Thin Clients Writing to Leostream API section, select Do not
   use or set lower priority to all other protocols.

5. In the Command line parameters edit field, enter the command line parameters needed to connect
   the user with single sign-on.

   The default parameters, shown below, launch the Windows version of the VMware View client.

   ```
   --nonInteractive --serverURL {IP} --userName {USER} --password
   {PLAIN_PASSWORD} --domainName {DOMAIN} --desktopName {VM:NAME} --
   desktopProtocol PCOIP
   ```

   The Linux version of the VMware View client requires different parameter. If your users are logging
   in from a Linux client device, modify the command line parameters, as follows;

   ```
   --nonInteractive --serverURL (IP) --userName (USER) --password
   {PLAIN_PASSWORD} --domainName (DOMAIN) --desktopName (VM:NAME) --protocol
   PCOIP
   ```

   If you have users logging in from Windows and Linux devices, create two protocol plans and
   assign the appropriate plan based on the user’s location. See “Assigning Plans to Locations” in the
   Connection Broker Administrator’s Guide for more information.

6. In the Port for remote viewer check specify the port number that the Connection Broker pings to
determine if the desktop is available for PCoIP connections.

7. Click Save.
After you create your protocol plans, build a pool that contains only these virtual machines running the VMware View Direct-Connection Plugin. When creating a policy that uses this pool, ensure that you select the protocol plan that uses the VMware View client.

**Establishing Connections using the Leostream Web Client**

The Leostream Web client uses the VMware Horizon View client URI to launch a PCoIP connection to the desktop. To configure the Connection Broker to support PCoIP connections to virtual machines:

1. Create a pool of virtual machines with a running VMware Horizon View Agent Direct-Connection Plug-In.
2. Create a protocol plan to assign to these virtual machines. In the **Web Browser** section of the protocol plan:
   a. Set the **Priority** of the **External viewer** to 1.
   b. Set the **Priority** of all other protocols to **Do not use**.
   c. In the **Configuration file** for the external viewer, enter:

   ```
   vmware-view://{HOSTNAME}/{VM:NAME}?desktopProtocol=PCOIP
   ```
3. Build a policy that assigns the protocol plan from step 2 to the pool of virtual machines created in step 1.
4. Assign the policy to the user.

When a user who is assigned this policy logs into the Connection Broker, the broker offers the user a virtual machine from the pool. When the user requests a connection to the virtual machine, the Connection Broker launches the VMware Horizon View client, which establishes the PCoIP connection to the desktop.

*The VMware Horizon View client URI does not support single sign-on.*

**Establishing Connections using a PCoIP Zero Client**

Users using a PCoIP zero client can connect to workstations and virtual machines by using the PCoIP Broker Protocol.

*By default, when the Connection Broker discovers new PCoIP zero clients, the broker configures the client to use the Connection Management Interface. You must manually switch the client to the PCoIP Broker Protocol, as follows.*
1. Go to the zero client’s **Configuration** dialog or the **Configuration** menu in the client’s Web interface

2. Select **Session**.

3. In the **Session** page:
   
a. Select **PCoIP Connection Manager** from the **Session Connection Type** drop-down menu

   b. Enter your Leostream Connection Broker address in the **Server URI** edit field. For example:

      ![Session Configuration](image)

      **Session**
      
      Configure the connection to a device

      **Session Connection Type:** PCoIP Connection Manager
      **Server URI:** https://broker.yourcompany.net

4. Click **Apply**.

When using a PCoIP zero client to connect to virtual machines, the Connection Broker ignores the protocol plan selected in the user’s policy and, instead, always establishes a PCoIP connection when the virtual machine has a running VMware Horizon View Direct-Connection Plugin.
rdesktop RDP Remote Viewer

You can use the rdesktop open source RDP remote viewer to connect to Windows desktops from a Linux client. To configure a protocol plan to use rdesktop:

1. In the protocol plan, scroll down to the rdesktop section.

2. Change the rdesktop **Priority** to 1 to make rdesktop the primary protocol for the Connection Broker to use or select a lower priority to use rdesktop as a backup protocol.

   If your protocol plan assigns priorities to multiple protocols, you must ensure that rdesktop has a higher priority than RDP. All three of these protocols use the same port. Therefore, the Connection Broker uses whichever protocol has the highest priority without trying the other two protocols.

3. Use the **Command line parameters** field to customize the remote viewer. The default command line parameters are:

   `-u {USER} -p {PLAIN_PASSWORD} -d {DOMAIN} {IP} -f`

   Remove the `-f` option for users that need access to the Leostream Connect Sidebar menu. When in full screen mode, rdesktop forces the remote desktop window to the top, hiding the Sidebar.

   You can use any rdesktop command line option, such as `-f` for full screen mode. See the rdesktop [documentation](#) for a description of supported command line parameters.

   To use rdesktop in conjunction with the Java version of Leostream Connect running Apple macOS, you must recompile rdesktop. Consult the FAQ in the Leostream Web site for more information.

VNC Remote Viewer

Leostream Connect supports the following versions of VNC; RealVNC®, RealVNC Enterprise, TigerVNC, TightVNC, and UltraVNC. UltraVNC allows the Windows username and password to be sent, enabling single sign-on.

To support single sign-on to a remote Windows desktop connected via VNC, you must select the single sign on task when installing the Leostream Agent on the remote desktops. Also, these users must have the **Enable single-sign-on to desktop console** option selected in the When User is Assigned to Desktop section of their policy.

Setting up the Connection Broker to Use VNC

To configure a protocol plan to use VNC

1. Scroll down to the **VNC** section of the protocol plan.
2. Select 1 from the Priority drop-down menu.

3. Use the Command line parameters and Configuration file fields to customize the remote viewer session.

⚠️ The default configuration file includes a dynamic tag for the VNC port, which the Connection Broker automatically replaces with the correct port when using NICE DCV. For VNC connections, you must specify the port for the VNC session. By default, the VNC port is 5900 when connection to a Windows desktop. If using VNC to connect to a Linux desktop, the port is 5901.

When a user requests a DCV connection to a desktop, the Connection Broker contacts the Leostream Agent running on the desktop to start the underlying VNC session for the user. After starting the session, the Leostream Agent returns the port associated with the session to the Connection Broker.

New protocol plans use the {PORT} dynamic tag in the configuration file. The Connection Broker replaces this tag the correct port for the user’s DCV connection before sending the VNC-file to the client device.

**VNC Command Line Parameters**

You can customize the VNC session using command line settings entered in the Command line parameters field. The command line parameters have the following format:

```
{IP}:nnnn [other_options]
```

Where:

- `{IP}`: The IP address completed by the Connection Broker.
- `:nnnn`: The port.

- `-listen [port]`
  Make the viewer listen on the given port for reverse connections from a VNC server. If no port is supplied, the command defaults to port 5500. WinVNC supports reverse connections using the Add New Client menu option, or the -connect command line option. Xvnc requires the use of the helper program vncconfig.

- `-via gateway`
  Automatically create encrypted TCP tunnel to the gateway machine before connection, connect to the host through that tunnel (TightVNC-specific). By default, this option invokes SSH local port forwarding, assuming that SSH client binary can be accessed as /usr/bin/ssh. Note that when using the -via option, the host machine name should be specified as known to the gateway machine, e.g. localhost denotes the gateway, not the machine where vncviewer was launched. See the ENVIRONMENT section below for the information on configuring the -via option.

- `-shared`
  When connecting, specify that a shared connection is requested. If this option is not set, when you make a connection, all other existing connections are closed. In TightVNC, this option is on, by default, allowing you to share the desktop with other clients already using it.
-noshared
When connecting, specify that the session may not be shared. This would either disconnect other connected clients or refuse your connection, depending on the server configuration.

-viewonly
Disable transfer of mouse and keyboard events from the client to the server. Often used in conjunction with -shared.

-fullscreen
Start in full-screen mode. Operating in full-screen mode may confuse X window managers. Typically, such conflicts cause incorrect handling of input focus or make the viewer window disappear mysteriously. See the grabKeyboard setting in the RESOURCES section below for a method to solve input focus problem.

-noraiseonbeep
By default, the viewer shows and raises its window on remote beep (bell) event. This option disables such behavior (TightVNC-specific).

-user username
User name for UNIX® login authentication. Default is to use current UNIX user name. If this option is given, the viewer prefers UNIX login authentication over the standard VNC authentication.

-passwd passwd-file
File from which to get the password (as generated by the vncpasswd(1) program). The file is typically stored in ~/.vnc/passwd. This option affects only the standard VNC authentication and does not log the user in to Microsoft Windows.

-encodings encoding-list
TightVNC supports several different compression methods to encode screen updates. This option specifies a set of compression methods to use in order of preference. Specify encodings separated with spaces and enclosed in quotes, if more than one is specified. Available encodings, in default order for a remote connection, are copyrect tight hextile zlib corre rre raw. For a local connection (to the same machine), the default order to try is raw copyrect tight hextile zlib corre rre. Raw encoding is always assumed as a last option if no other encoding can be used for some reason. For more information on encodings, see the section ENCODINGS below.

-bgr233
Always use the BGR233 format to encode pixel data. This reduces network traffic, but colors may be represented inaccurately. The bgr233 format is an 8-bit true color format, with 2 bits blue, 3 bits green, and 3 bits red.

-owncmap
Try to use a PseudoColor visual and a private colormap. This allows the VNC server to control the colormap.

-truecolour, -truecolor
Try to use a TrueColor visual.

-depth depth
On an X server which supports multiple TrueColor visuals of different depths, attempt to use the specified one (in bits per pixel). If successful, this depth is requested from the VNC server.
-compresslevel level
Use specified compression level (0 to 9) for "tight" and "zlib" encodings (TightVNC-specific). Level 1 uses minimum of CPU time and achieves weak compression ratios, while level 9 offers best compression but is slow in terms of CPU time consumption on the server side. Use high levels with very slow network connections, and low levels when working over high-speed LANs. Do not use compression level 0; start with the level 1.

-quality level
Use the specified JPEG quality level (0 to 9) for the "tight" encoding (TightVNC-specific). Level 0 denotes bad image quality but impressive compression ratios, while level 9 offers good image quality at lower compression ratios. The "tight" encoder uses JPEG to encode only those screen areas that are suitable for compression that experiences loss, so quality level 0 does not always mean unacceptable image quality.

-nojpeg
Disable JPEG compression that experiences loss in tight encoding (TightVNC-specific). Disabling JPEG compression is not a good idea in typical cases, as the tight encoder becomes less efficient. Use this option if it is absolutely necessary to achieve perfect image quality (see also the -quality option).

-nocursorshape
Disable cursor shape updates, protocol extensions used to handle remote cursor movements locally on the client side (TightVNC-specific). Using cursor shape updates decreases delays with remote cursor movements, and can improve bandwidth usage dramatically.

-x11cursor
Use a real X11 cursor with X-style cursor shape updates, instead of drawing the remote cursor on the framebuffer. This option also disables the dot cursor, and disables cursor position updates in non-fullscreen mode.

-autopass
Read a plain-text password from stdin. This option affects only the standard VNC authentication.
Session Shadowing and Collaboration

The Connection Broker allows users that connect to desktops using the Mechdyne TGX, NoMachine, and HP RGS protocols to collaborate by inviting a shadow user to connect to their session.

Configuring Collaboration in the Connection Broker

In order to collaborate, a user must have the appropriate role and policy setting, as described in the following procedure. The role determines which users have permission to share or shadow sessions. The policy determines which specific desktops support collaboration.

1. **Build pools:** To simplify creating policies, construct pools that contain only desktops that support collaboration.

2. **Configure roles:** You must explicitly give permission to each user that is either sharing their remote session or shadows another user’s session. To provide the necessary permission, select the **Allow user to collaborate with other users** option in the user’s role, as shown in the following figure.

3. **Create policies:** In addition to giving the user permission to collaborate with other users, you must indicate which remote sessions support collaboration. Use settings in the user’s policies to indicate which sessions support collaboration, as follows.
   
   a. Go to the > **Configuration > Policies** page.
   
   b. Edit the user’s policy or create a new policy.
   
   c. In the **Desktop Assignments from Pool** section, select a pool that contains desktops that support collaboration.
   
   d. In the **When User is Assigned to Desktop** section for the pool selected in step c, select the **Enable collaboration and session shadowing** option, as shown in the following figure.
e. For NoMachine sessions, by default, shadowed sessions allow the shadow user to interact with the session. To restrict the shadowed session to be view-only, select the **View only shadowing, not interactive** option. The interactivity of HP RGS sessions is controlled by settings in the RGS Sender.

4. **Define assignments**: After configuring a role and policy that support collaboration, you must configure the > **Configuration > Assignments** pages to assign that role and policy to the appropriate users.

**Working with Invitations in the Leostream Web Client**

**Sending a Collaboration Invitation**

A user with permission to invite other users to shadow their remote session sends the invitation, as follows.

1. Click the **Connect** link to establish the desktop connection.

2. After the sessions starts, click the **Refresh List** link in the Leostream Web client. All desktops with an assigned desktop that allows collaboration includes a new **Invite** link.

3. Click the **Invite** link, shown in the following figure.

![Image of inviting process](image)

1. In the **Send Invitation** form that opens, select the user to invite to the session from the **Shadow user** drop-down menu.
Only Leostream users who are already listed on the Connection Broker > Users page can be entered as a shadow user. The selected user must have a role that gives them permission to collaborate with other users. Otherwise, the user will not see the invitation.

4. From the **Expire after** drop-down menu, indicate when the invitation should expire. Leave this drop-down menu blank if the invitation should be valid for the lifetime of the remote session.

5. In the **Notes** field, provide an optional message to display to the invited user, for example:

![Send Invitation](image)

6. Click **Send**.

The Leostream Web client now displays the new invitation, for example:

![Resources](image)

Select **Invitations** from the **Show** menu to limit the resource list to only sent and received invitations, as shown in the following figure.
Click the **View** link associated with an invitation to see the details of that information, including any entered notes, for example:

**Cancelling an Invitation**

To cancel an invitation, click the **Cancel** link associated with that invitation, as shown in the following figure.
Invitations are automatically cancelled when the desktop is released back to its pool, for example, when you log out of the remote session and have a release plan that releases the desktop on logout.

**Accepting a Collaboration Invitation**

A user’s pending invitations appear in the Resource list with other offered desktops, as shown in the following figure.

- Click the **View** link to see the invitation’s details, including any message sent with the invitation.
- Click the **Decline** link to remove the invitation from the resource list without connecting to the shadowed session.
- Click the **Connect** link to shadow the session.

**Working with Invitations using Leostream Connect**

**Sending a Collaboration Invitation**

After logging into the Windows version of Leostream Connect and launching their remote session, a user with permission to invite other users to shadow their session can invite the user, as follows.

2. After the remote sessions starts, right-click on the Leostream Connect System Tray menu and select the **Refresh List** option. All desktops with an assigned desktop that allows collaboration includes a new **Invite** link.

3. Again, right-click on the Leostream Connect System Tray menu and highlight the desktop that you want to invite the user to shadow. When you highlight the desktop name, a submenu opens.

4. Select **Invite**... from the submenu. The **Send Invitation** form opens.

5. In the **Shadow user** field, begin typing the name of the user to invite.

Only Leostream users who are already listed on the Connection Broker > **Users** page can be entered as a shadow user. The selected user must have a role that gives them permission to collaborate with other users. Otherwise, the user will not see the invitation.
6. From the **Expire after** drop-down menu, indicate when the invitation should expire. Leave this drop-down menu blank if the invitation should be valid for the lifetime of the remote session.

7. In the **Notes** field, provide an optional message to display to the invited user.

8. Click **Send** to send the invitation or **Cancel** to close the form without sending the invitation.

**Viewing and Cancelling Invitations**

After you send an invitation, right-click on the Leostream Connect System Tray menu and select the **Refresh List** option. A new **View Invitations** menu items is added to the list.

Selecting the **View Invitations** menu opens up a dialog displaying the list of invitations that have been sent by you.

Click the **Cancel** link to cancel any invitations you sent to other users.

**Accepting a Collaboration Invitation**

To access an invitation sent to you from another user, log into Leostream Connect and then close the **Connect** dialog. Right click on the Leostream Connect System Tray menu and select **View Invitations**. The Invitations list dialog that opens shows all the pending invitations for the logged in user.

- Click the **Decline** link to remove the invitation from the resource list without connecting to the shadowed session.

- Click the **Connect** link to shadow the session.

**Managing Invitations in the Connection Broker**

All past and present invitations appear on the > **Resources** > **Collaboration** page, shown in the following figure.
The **Actions** links allow you to do the following:

- Click **View** to see details about the invitation.
- Click **Cancel** to cancel the invitation

The **Status** column provides information about what actions have been taken on the initiation.

- **Cancelled** indicates that the invitation was cancelled. The **Viewed** column indicates if the invited user connected to the shadowed session before the invitation was cancelled.
- **Declined** indicates that the invitee declined the invitation.
- **Expired** indicates that the invitation has expired and the invitee can no longer join the session.
- **Sent** indicates an invitation has been sent, but has not been declined or viewed by the invitee.
- **Viewed** indicates that the invitee has connected to the shadowed session.
User Configurable Protocol Plan Parameters

User-configurable protocol plan parameters give users control over the look-and-feel of their desktop connection when using NoMachine NX and HP RGS display protocols.

Defining Scope of the Configured Parameter

User-configurable values are stored either globally and used for all connections made by that user, or individually for a desktop/client pair.

The scope of the user-configured values is determined by the setting of the Store user-configured protocol parameters drop-down menu in the user’s policy, shown in the following figure.

- **Globally for all connections and from every client** allows the user to set the value once and use it everywhere. In this case, the setting applies to all policies, pools, and desktops assigned to the user.

- **Individually for each connection/client pair** allows the user to configure different values for each of their connections. In this case, the user must reset their desired value when they log into Leostream at a different client device.

The Connection Broker considers Leostream Connect and the Leostream Web client accessed from the same physical device as two different clients.
End-User Interface for Configuring Parameters

End-users can configure protocol plan parameters when logging in through the Leostream Web client and Leostream Connect.

**Leostream Web Client**

Users logging in from the Leostream Web client set user-configurable parameters, as follows.

1. After logging into the Web client, any connections with configurable parameters include a Configure link, as shown in the following figure.

![Configure link](image)

2. After the user clicks the Configure link, a new form opens, displaying the parameters they are allowed to set, for example:

3. Clicks Save and Close to store new values for all subsequent applicable connections. Click Apply and Close to use the new values only for the current connection.
Leostream Connect

Users logging in from Leostream Connect set protocol parameters, as follows.

1. After logging into the Leostream Connect client, highlighting any connection with configurable parameters enables a Configure button at the bottom-right of the dialog.

   The Java version of Leostream Connect uses a JavaFX based browser component to display the dialog. You must install a JRE that includes JavaFX to configure parameters using the Java version of Leostream Connect.

2. After the user clicks the Configure link, a new form opens, displaying the parameters they are allowed to set.

3. Clicks Save and Close to store new values for all subsequent applicable connections. Click Apply and Close to use the new values only for the current connection.

Setting Global User-Configurable Parameters

When the user’s policy is set to store user-configure parameters globally, the same parameter values are used regardless of which policy, pool, or desktop the user is currently offered. For example, even if the user’s policy changes based on their location, the same configured values are used at each location.

In addition, the Connection Broker stores global parameters based on the protocol. Therefore, all connections that use the HP RGS display protocol, for example, use the same global parameter set.

If the policy specifies that user-configured parameters are stored globally, the user is prompted to set the parameters for all connections offered by a policy when they click the Configure link for a particular connection.

If users need more flexibility when setting protocol plan parameters, modify their policy to store parameters individual for each desktop/client pair.