



## **Connection Broker**

**Managing User Connections to Workstations, Blades,  
VDI, and More**

## **Choosing and Using Display Protocols**

October 21, 2015

## Contacting Leostream

Leostream Corporation  
465 Waverley Oaks Rd.  
Suite 200  
Waltham, MA 02452  
USA

<http://www.leostream.com>  
Telephone: +1 781 890 2019  
Fax: +1 781 688 9338

To submit an enhancement request, email [features@leostream.com](mailto:features@leostream.com).

To request product information or inquire about our future direction, email [sales@leostream.com](mailto:sales@leostream.com).

## Copyright

© Copyright 2002-2015 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. Sun, Sun Microsystems, Sun Ray, and Java are trademarks or 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

<b>CONTENTS</b>	<b>3</b>
<b>OVERVIEW</b>	<b>5</b>
<b>CHOOSING A DISPLAY PROTOCOL</b>	<b>5</b>
THE PROTOCOL TRIANGLE	5
A QUESTION OF OPERATING SYSTEM	7
<b>CONFIGURING DISPLAY PROTOCOLS IN LEOSTREAM</b>	<b>8</b>
USING PROTOCOL PLANS	8
<i>How Protocol Plans Work</i>	8
<i>Building Protocol Plans</i>	9
SPECIFYING CONFIGURATION FILES AND COMMAND LINE ARGUMENTS	12
<i>Using Dynamic Tags in Configuration Files</i>	12
<i>Dynamic Remapping of Desktop IP Address</i>	16
<i>Client Dependent Variables</i>	17
<b>CITRIX HDX</b>	<b>18</b>
CONFIGURING THE CITRIX XENDESKTOP ENVIRONMENT	19
<i>Configuring the Citrix XenDesktop Studio or Citrix Studio</i>	19
<i>Creating a Citrix XenApp Services Site in XenDesktop 5</i>	20
<i>Setting up a Citrix Storefront in XenDesktop 7</i>	20
CONFIGURING CLIENT DEVICES	21
HDX CONNECTIONS TO PERSISTENT DESKTOPS ASSIGNED BY LEOSTREAM	21
<i>Step 1: Create a Citrix XenDesktop Center</i>	21
<i>Step 2: Define an HDX Protocol Plan</i>	23
<i>Step 3: Create a Pool of HDX-Enabled Desktops</i>	24
<i>Step 4: Use the HDX Protocol Plan in Policies</i>	24
HDX CONNECTIONS TO NON-PERSISTENT DESKTOPS ASSIGNED BY LEOSTREAM	25
HDX CONNECTIONS TO RESOURCES ASSIGNED BY CITRIX	25
<i>Step 1: Enable the Citrix XenApp Services Site Feature</i>	25
<i>Step 2: Configure the Policy</i>	26
<b>CITRIX ICA</b>	<b>26</b>
USING THE CITRIX ONLINE PLUGIN	27
<i>Launching Desktop Connections in Fullscreen</i>	28
USING THE CITRIX CLIENT FOR JAVA	29
<i>Uploading New Client Versions</i>	30
<i>Launching the Client in a new Window</i>	30
<b>ERICOM® BLAZE</b>	<b>30</b>
<b>EXCEED ONDEMAND</b>	<b>32</b>
<b>FAMATECH RADMIN® 2.2 AND 3.X REMOTE VIEWER</b>	<b>34</b>
<b>HP® REMOTE GRAPHICS SOFTWARE (RGS)</b>	<b>34</b>
<i>HP RGS Protocol Plan Options</i>	35
<i>Multi-Monitor Support with HP RGS</i>	36
<i>Activating HP Velocity and Advanced Video Compression Features</i>	36

<i>Remembering Window Position for HP RGS Connections</i> .....	37
<i>Single Sign-On with HP RGS</i> .....	38
<i>Setting User Configurable HP RGS Parameters</i> .....	39
<i>USB Passthrough with HP RGS</i> .....	41
<i>Using HP SAM Clients</i> .....	42
<b>MICROSOFT® RDP AND REMOTEFX</b> .....	<b>44</b>
<i>Options for Encoding Desktop Login Credentials into RDP Configuration Files</i> .....	45
<i>Microsoft RDP Viewer Command Line Parameters</i> .....	45
<i>Microsoft RDP Viewer Configuration File Variables</i> .....	45
<b>NOMACHINE NX</b> .....	<b>53</b>
LAUNCHING NX CONNECTIONS FROM THE WEB CLIENT .....	54
NX CONFIGURATION FILE .....	55
UPGRADING THE NX JAVA APPLET .....	55
SETTING USER-CONFIGURABLE NX PARAMETERS .....	56
SESSION SHADOWING AND COLLABORATION.....	58
<i>Configuring Collaboration in the Connection Broker</i> .....	58
<i>Managing Shadowed Sessions in the Connection Broker</i> .....	60
<i>Using NoMachine NX Collaboration in the Leostream Web Client</i> .....	60
<b>TERADICI® PCOIP® TECHNOLOGY</b> .....	<b>63</b>
ENABLING PCOIP REMOTE WORKSTATION CARD SUPPORT .....	63
PCOIP CONNECTIONS TO WORKSTATIONS WITH A REMOTE WORKSTATION CARD .....	64
PCOIP CONNECTIONS USING THE PERVASIVE COMPUTING PLATFORM.....	64
PCOIP CONNECTIONS TO VMWARE VIRTUAL MACHINES .....	64
<i>Establishing Connections using Leostream Connect</i> .....	65
<i>Establishing Connections using the Leostream Web Client</i> .....	66
<i>Establishing Connections using a PCoIP Zero Client</i> .....	67
<b>RED HAT SPICE</b> .....	<b>68</b>
CONFIGURING THE CLIENT DEVICE .....	68
<i>Installing the SPICE Client</i> .....	68
CONFIGURING A CONNECTION BROKER PROTOCOL PLAN FOR SPICE .....	69
<b>RDESKTOP RDP REMOTE VIEWER</b> .....	<b>70</b>
<b>SUN RAY OPTIONS</b> .....	<b>70</b>
SUN RAY – UTTSC.....	70
SUN SECURE GLOBAL DESKTOP – TTATSC .....	71
<b>VNC REMOTE VIEWER</b> .....	<b>71</b>
<i>Setting up VNC for Single Sign-On on Windows Operating Systems</i> .....	72
<i>Setting up the Connection Broker to Use VNC</i> .....	74
VNC Command Line Parameters .....	75
<i>RealVNC Enterprise Edition, UltraVNC, and TightVNC Configuration file</i> .....	77
<b>USER CONFIGURABLE PROTOCOL PLAN PARAMETERS</b> .....	<b>78</b>
END-USER INTERFACE FOR CONFIGURING PARAMETERS.....	78
<i>Leostream Web Client</i> .....	78
<i>Leostream Connect</i> .....	79
SETTING GLOBAL USER-CONFIGURABLE PARAMETERS.....	80

## Overview

The Leostream Connection Broker supports a wide range of display protocols that allow you to tailor your environment and provide the end-user experience required throughout your entire organization. The Leostream Connection Broker currently supports the following display protocols

- Citrix HDX and ICA
- Ericom Blaze
- Famatech Radmin
- HP Remote Graphics Software (RGS)
- Microsoft RDP, including ActiveX RDP for Web browser connections
- Microsoft RemoteFX
- NoMachine NX
- Red Hat SPICE
- rdesktop
- OpenText Exceed onDemand
- Oracle Appliance Link Protocol (ALP)
- Oracle Adaptive Internet Protocol (AIP)
- Teradici PCoIP
- VNC (RealVNC, TightVNC, and UltraVNC)

This document describes each of these protocols in a separate chapter.

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 can build your protocol plans, however, you must choose the display protocols you will use in your environment. The next chapter, [\*\*Choosing a Display Protocol\*\*](#), 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. Once the connection is established, the Connection Broker removes itself from the connection path, i.e., the Connection Broker does not proxy the remote session.

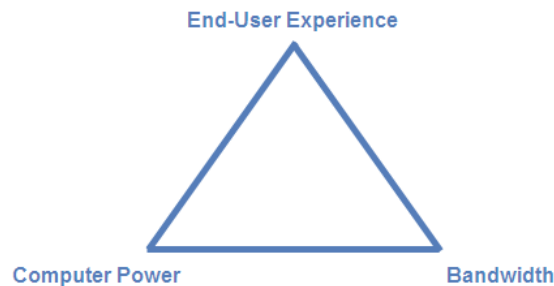
The performance and requirements of the remote session are, therefore, completely determined by the display protocol you select. This chapter tries to provide some food-for-thought when investigating and choosing from the virtual soup of display protocols.

## The Protocol Triangle

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

- Low bandwidth
- 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 always has repercussions for the other angles.



You can typically achieve any two of the previous goals, but you 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 must have 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.

- 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 (see [A Question of Operating System](#))? For example, not all display protocols support Linux operating systems on the backend.
- If you are using thin clients, which display protocols does it natively support, and are there other protocols the thin client can be manually configured to support?
- Are your users accessing an entire desktop or only an application?
- Is single sign-on a requirement, or just nice-to-have?
- How large will your deployment grow? (High compute power may lower the hosting environment's scalability.)

- Do you have users that connect to workstation/blades that provide a lot of native compute power?

## A Question of Operating System

Your display protocol choice is also influenced by the types of operating systems you run on your remote desktops, as well as on your client devices. The following table shows which display protocols can be used to connect to and from either Linux or Windows operating systems.

Display Protocol	Client OS	Connect to Linux OS	Connect to Windows OS
Citrix HDX	Linux	No	Yes
	Windows	No	Yes
Ericom Blaze	Linux	No	Yes
	Windows	No	Yes
Exceed onDemand	Linux	Yes	No*
	Windows	Yes	No*
Famatech Radmin	Linux	No	Yes**
	Windows	No	Yes
HP RGS	Linux	Yes	Yes
	Windows	Yes	Yes
Microsoft RemoteFX/RDP	Linux	No	No
	Windows	No	Yes
NoMachine NX	Linux	Yes	No***
	Windows	Yes	No***
rdesktop	Linux	No	Yes
	Windows	No	No
	Windows		
VNC	Linux	Yes	Yes
	Windows	Yes	Yes

\* Although the Exceed onDemand protocol supports connecting to Windows operating systems, the Leostream Agent for Windows operating systems does not manage Exceed onDemand sessions.

\*\* The Radmin Viewer is compatible with the Wine software for running Windows applications on a Linux client.

\*\*\* NX Server support for Windows operating systems is under development.

For thin clients running a vendor-supplied operating system, please consult your thin client vendor.

# Configuring Display Protocols in Leostream

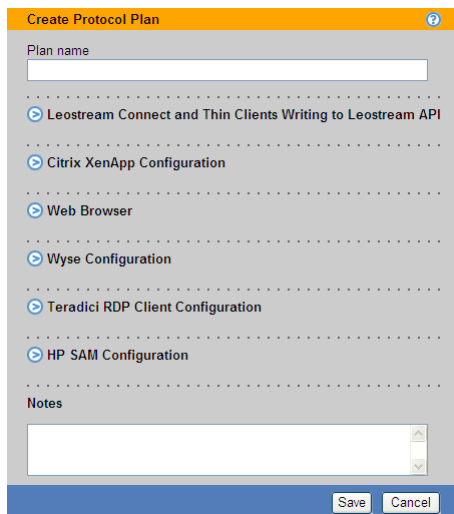
## Using Protocol Plans

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 **> Plans > Protocol** 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 zero 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, shown in the following figure.



## 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

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

The screenshot shows the 'Create Protocol Plan' dialog box. It has a title bar 'Create Protocol Plan' with a help icon. Below is a 'Plan name' text field. A section titled 'Leostream Connect and Thin Clients Writing to Leostream API' is expanded. It contains three protocol configurations:


- RDP**: Priority: 1 (dropdown). Command line parameters: (empty text field). Configuration file: screen mode id:1:2, desktopwidth:1:1024, desktopheight:1:768 (text area with scrollbars).
- RGS**: Priority: 2 (dropdown). Configuration file: (empty text area).
- VNC**: Priority: Do not use (dropdown).

Red arrows and text annotations are present:

- An arrow points to the 'Leostream Connect and Thin Clients Writing to Leostream API' section header with the text: 'Each section configures the remote viewers for a particular client device.'
- An arrow points to the RDP Priority dropdown with the text: 'The Priority determines the order in which the Connection Broker should try to use each remote viewer.'
- An arrow points to the RDP Configuration file text area with the text: 'Command line parameters and configuration files determine exactly how the connection is established.'

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, 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.

## 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 remote viewing protocol? If not, which remote viewing protocols will they use? If these remote viewers communicate over the same port, you will need a protocol plan for each remote viewer.
- For each remote viewer 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 of command line parameters and configuration file.
- Do your remote desktops support multiple remote viewers, such as RDP, RGS, and VNC? If so, and you want to allow different users to access different remote viewers, you will need a protocol plan that defines the appropriate priorities for each remote viewer.

The above questions are examples of the 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 **> Plans > Protocols** 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 following figure, configure the remote viewers 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 laptop or fat client
  - A thin client with an installed Leostream Connect client
  - A thin client with a customized Leostream client

**Leostream Connect and Thin Clients Writing to Leostream API**

**RDP and RemoteFX** Priority: 1

Command line parameters

Configuration file

```
screen mode id:i:2
desktopwidth:i:1024
desktopheight:i:768
```

**RGS** Priority: Do not use

This section selects the protocols to use when the user logs in through Leostream Connect or any thin client that writes to the Leostream API.

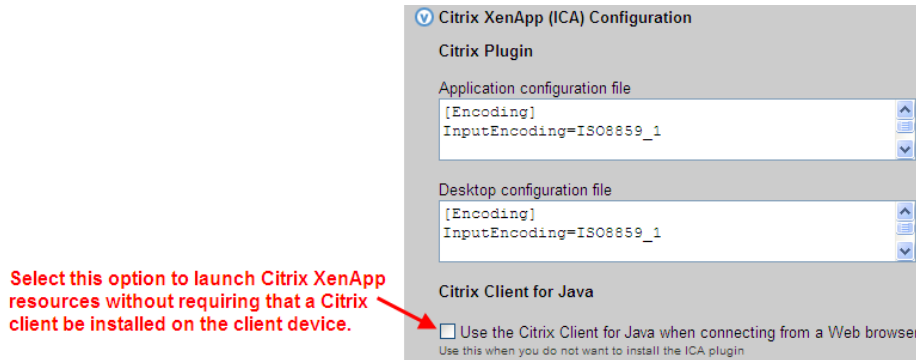
Specify the Command line parameters and/or Configuration file to use to launch the remote viewer.

The Priority indicates the order in which the Connection Broker tries to launch the remote viewers. If you specifically do not want to use a particular protocol, select "Do not use".

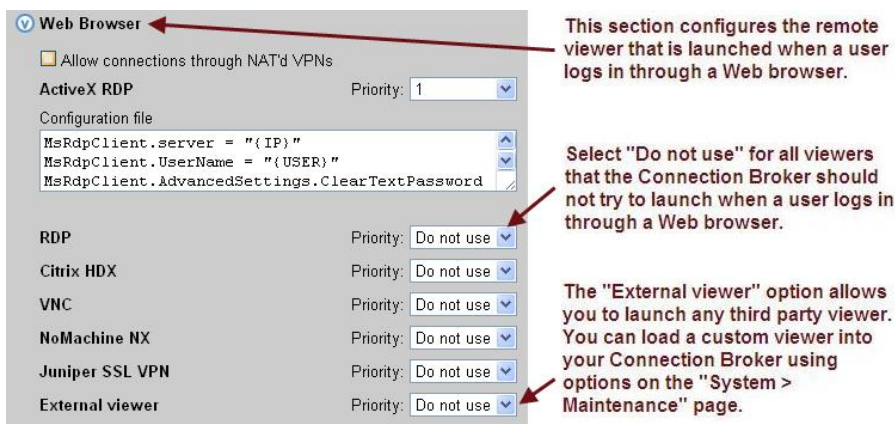
5. In the **Citrix XenApp (ICA) Configuration** section, shown in the following figure, configure the command line parameters and ICA-file to use when launching a desktop or application published in

a Citrix XenApp farm. This section applies to users logging in from any of the following client devices

- The Windows or Java version of Leostream Connect
- The Leostream Web client.



6. In the **Web Browser** section, shown in the following figure, configure the remote viewer to use when a user logs in through the Leostream Web client.



7. Configure the remainder of the protocol plan, shown in the following figure, if your end users log in through any of the following client devices.

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

**Wyse Configuration**

Desktop configuration file - {USER}.ini

```
connect=rdp
autoconnect=yes
host={ IP }
```

Application configuration file - {USER}.ini

```
connect=ica
application={ CITRIX_RESOURCE }
autoconnect=no
```

---

**Teradici PCoIP Client Configuration**

Alternate port for remote viewer port check

8080

If policies use the remote viewer port check to invoke backup pools, enter the port to check for PCoIP connections

---

**HP SAM Configuration**

Configuration file

```
<OffsetX>0</OffsetX>
<OffsetY>0</OffsetY>
<X>0</X>
```

8. Use the **Notes** field to store any additional information with your protocol plan.
9. Click **Save** to store any changes to the plan.

## Specifying Configuration Files and Command Line Arguments

The configuration file 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 allow you to customize certain display protocol behaviors. The Connection Broker supports dynamic tags in the **Command line parameters** and **Configuration file** fields for any of the protocol. When establishing a remote session, the Connection Broker replaces dynamic tags with the appropriate information.

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 does not alter the text in the configuration file.

Dynamic Tags	Purpose
{ IP }	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.

Dynamic Tags	Purpose
{IP_ADDRESS}	The IP address of the desktop or, in the case of ICA connections, the IP address of the Citrix XenApp farm that publishes the desktop or application specified by the dynamic tag {CITRIX_RESOURCE}.
{HOSTNAME}	The hostname of the desktop or, in the case of ICA connections, the hostname of the Citrix XenApp farm that publishes the desktop or application specified by the dynamic tag {CITRIX_RESOURCE}.
{IP_ADDRESS-or-HOSTNAME}	The IP address of the desktop or, if the IP address is not available, the hostname of the desktop.
{HOSTNAME-or-IP_ADDRESS}	The hostname of the desktop or, if the hostname is not available, the IP address of the desktop.
{SHORT_HOSTNAME}	The short hostname of the desktop, or the hostname cut at the first dot. For example, if the hostname is <code>desktop.example.com</code> , the {SHORT_HOSTNAME} tag returns <code>desktop</code> .
{USER}, {USER:USER}, {USER:LOGIN_NAME}, or {LOGIN) NAME}	The user's login name. This value corresponds to the value shown in the <b>Login name</b> column on the > <b>Users &gt; Users</b> page. To force the login name on the remote desktop to upper or lower case, include the <code>:lowercase</code> or <code>:uppercase</code> modifier, for example {USER:lowercase} or {USER:LOGIN_NAME:uppercase}.
{AD:USER:attribute_name}	The value found in the user's Active Directory attribute given by <i>attribute_name</i> . 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 Chapter 10 of the <b>Connection Broker Administrator's Guide</b> .
{NAME} or {USER:NAME}	The user's display name. This value corresponds to the value shown in the <b>Name</b> column on the > <b>Users &gt; Users</b> page.
{AD_DN} or {USER:AD_DN}	The user's Active Directory Distinguished Name. This value corresponds to the value shown in the <b>AD Distinguished Name</b> column on the > <b>Users &gt; Users</b> page.
{EMAIL} or {USER:EMAIL}	The user's email address. This value corresponds to the value shown in the <b>Email</b> column on the > <b>Users &gt; Users</b> page.
{PRE_EMAIL} or {USER:PRE_EMAIL}	The portion of the user's email address before the @ symbol.
{POST_EMAIL} or {USER:POST_EMAIL}	The portion of the user's email address after the @ symbol.

Dynamic Tags	Purpose
{DOMAIN}	The name entered into the <b>Domain</b> field for the authentication server that authenticated a user. If the <b>Domain</b> field is empty, the Connection Broker replaces this dynamic tag with the value entered or selected in the <b>Domain</b> field of the user's client.
{AUTH_DOMAIN}	The name entered in the <b>Authentication server name</b> field of the authentication server that authenticated the current user.
{PLAIN_PASSWORD}	The user's password, in plain text
{RDP_PASSWORD}	For Leostream Connect, the user's password encrypted for RDP usage
{SCRAMBLED_PASSWORD}	For NoMachine NX client, only, the user's password is scrambled to prevent casual eavesdropping
{STANDARD_RDP_PASSWORD:xxxx}	For Leostream Connect, a specific password encrypted for RDP usage
{HOST:IP}	For use in the SPICE command line parameters, resolves to the IP address of the Red Hat Enterprise Virtualization environment that manages the virtual machine.
{HOST:PORT}	For use in the SPICE command line parameters, resolves to the port used to establish a SPICE connect to the virtual machine.
{HOST:SECURE_PORT}	For use in the SPICE command line parameters, resolves to the secure port used to establish a SPICE connect to the virtual machine.
{SPICE_TICKET}	For use in SPICE command line parameters, the secure ticket needed to establish communication between the SPICE client and host.
{CLIENT} or {CLIENT:NAME}	The name of the user's client device used to log into the Connection Broker. This value corresponds to the value shown in the <b>Name</b> column on the > <b>Clients</b> > <b>Clients</b> page.
{CLIENT:IP}	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 <b>IP Address</b> column on the > <b>Clients</b> > <b>Clients</b> page.
{CLIENT:MAC}	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 <b>MAC Address</b> column on the > <b>Clients</b> > <b>Clients</b> page.
{CLIENT:TYPE}	The type of client used to log into the Connection Broker. This value corresponds to the value shown in the <b>Type</b> column on the > <b>Clients</b> > <b>Clients</b> page.
{CLIENT:MANUFACTURER}	The manufacturer of client used to log into the Connection Broker. This value corresponds to the value shown in the <b>Manufacturer</b> column on the > <b>Clients</b> > <b>Clients</b> page.

Dynamic Tags	Purpose
{CLIENT:UUID}	The UUID of the client used to log into the Connection Broker. This value corresponds to the value shown for the <b>Client UUID</b> on the <b>Edit Client</b> page.
{POOL:NAME}	The name of the pool that contains the desktop that the user is connecting to
{VM:NAME}	The name of the desktop the user is connecting to, as shown in the <b>Name</b> field on the <b>&gt; Resources &gt; Desktops</b> page.
{WINDOWS_NAME}	The guest host name of a desktop, as returned by the Leostream Agent
{FQDN}	If the user authenticated against an authentication server, the fully qualified name, e.g., <code>cn=Fred,ou=Users,o=Company</code>
{NOVELL_FQDN}	If user authenticated against an eDirectory authentication server this will be the fully qualified name in the format <code>.cn=Fred.ou=Users.o=Company</code>
{CITRIX_RESOURCE}	For ICA connections, the name of the published Citrix resource/application
{DRIVE:CD}	For the RDP configuration file, use <code>drivestoredirect:s:{DRIVE:CD}</code> to redirect all CD drives found on system. No other drives are directed.
{DRIVE:DVD}	For the RDP configuration file, use <code>drivestoredirect:s:{DRIVE:DVD}</code> to redirect all DVD drives found on system. No other drives are directed.
{LEO_SPAN}	For use with display plans, either 1 or 0 depending on if the RDP session should be spanned across multiple monitors.
{LOGOUT_URL}	The URL to log the user out of the session.
{LIST_URL}	The URL to view the list of desktops.
{ENV:*}	The value of the client side variable specified in *. So <code>{ENV: HTTP_COOKIE}</code> might return <code>uid=25157202</code> .
{REMAPPED_IP:X.X.X.X}	Re-maps IP addresses by replacing the non-X portion of the IP address with the specified tag.
{REMAPPED_IP:subnet_mask}	Re-maps IP addresses on different subnets.
{SESSION}	For use with the Java version of Leostream Connect. The session ID associated with session-based RGS Receiver configuration file parameters.
{USB_SESSION}	Indicates that the Java version of Leostream Connect should manage which remote RGS session has access to USB devices.

Dynamic Tags	Purpose
<code>{MATCHED_IP:partial_IP_address}</code>	<p>The IP address of the desktop, where <i>partial_IP_address</i> 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 <code>MATCHED_IP</code> 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 <code>{MATCHED_IP:10.110.1}</code> the IP address used for the connection is 10.110.1.14.</p> <p>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 <code>{MATCHED_IP:partial_IP_address-or-IP}</code>. If the desktop returns one IP addresses of 172.29.229.151 and the tag is <code>{MATCHED_IP:10.110.1-or-IP}</code> the IP address used for the connection is 172.29.229.151.</p> <p>When specifying <i>partial_IP_address</i>, trailing zeros are optional, e.g., <code>{MATCHED_IP:172.29.0.0}</code> is equivalent to <code>{MATCHED_IP:172.29}</code>.</p>

## Dynamic Remapping of Desktop IP Address

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

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

As an example, imagine an offshore development center than 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 NATed 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 supports specifying a network mask length and a target range for the source and destination.

The `{REMAPPED_IP:X.X.X.X}` syntax can be used 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.

## Client Dependent Variables

You can also use the client's IP address to determine if a particular client configuration variable is sent to the client.

For example, you can localize printing to enable the relevant printer to be mapped to the assigned virtual desktop.

You can also differentiate between users connecting locally and the same user connecting remotely. To do this, include logic statements in the client configuration file that have to be true for the embedded tag to

be included when the configuration file is download to the client.

These logic statements contain three parts. They can all be together, or on different lines. For example:

```
{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 has to 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 definitions spread across multiple lines.
- `{END_NETWORK}` closes the logic statement and must be present.

You can use multiple logic statements. The Connection Broker checks the statements in order and, as soon as one statement is true, the broker applies it and ignores the remaining statements.

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.

## Citrix HDX

The Connection Broker supports HDX connections when used in conjunction with Citrix XenDesktop 5.6 and XenDesktop 7.6 software.



To use Leostream to establish HDX connections you must separately obtain all necessary Citrix licensing. For information on XenDesktop licensing, contact your Citrix sales representative.

When using Leostream to manage HDX connections, the desktop assignments can be managed by either Leostream or Citrix.

1. If desktops are assigned to users via Leostream policies, Leostream *pushes* the assignments into the Citrix Studio (formerly Desktop Studio) in order to establish the HDX connection. Leostream can push assignments into XenDesktop versions 5.x and 7.x.



Leostream defines two types of desktops when working in a Citrix environment. A *persistent*

desktop is defined as a desktop whose operating system persists between reboots, while a *non-persistent* desktop has an operating system that is streamed down from Citrix Provisioning Services during a reboot. In this context, persistence applies to the operating system, not the user assignment.

- **Persistent** desktops that are assigned by Leostream must be inventoried using an Active Directory center.
  - **Non-persistent** desktops provisioned by Citrix Provisioning Server that are assigned by Leostream must be inventoried using the VMware vCenter Server center.
2. If desktops and applications are already assigned to users in Citrix Delivery Groups, Leostream *pulls* the assignments from a Citrix XenApp Services Site in order to offer those resources to users.

For more information on how Leostream integrates with Citrix XenDesktop, see the [Leostream Quick Start Guide for Citrix XenDesktop 7](#).

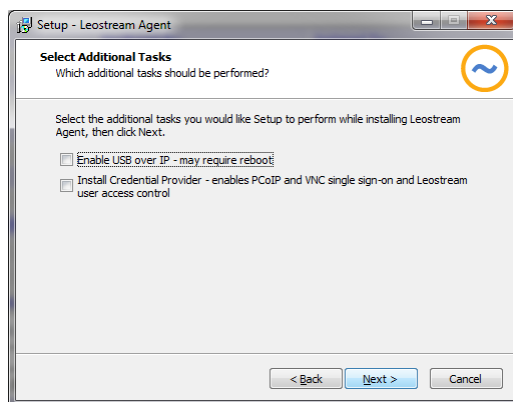
## Configuring the Citrix XenDesktop Environment

### Configuring the Citrix Desktop Studio or Citrix Studio

To have Leostream manage HDX connections to your XenDesktop environment, you must perform the following steps.

- Install a Leostream Agent on the server running Desktop Studio or Citrix Studio.

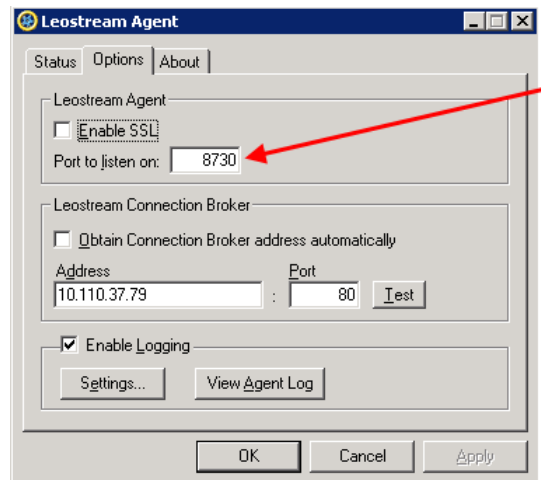
When installing the Leostream Agent, ensure that no additional features are installed, as shown in the following figure. For complete installation instructions, see the [Leostream Installation Guide](#).



After the Leostream Agent is installed, ensure that it communicates on a port that is different from all ports already in use by Citrix. Leostream recommends configuring the Leostream Agent to use port 8730, as follows.

1. On the Citrix Studio server, open the Leostream Agent Control Panel dialog.

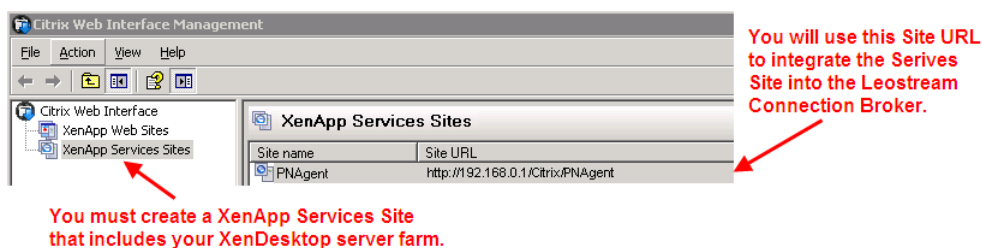
2. Go to the **Options** tab.
3. Change the **Port to listen on** to 8730, as shown in the following figure.



- Open the Citrix PowerShell prompt from the **Start** menu and ensure that the `Get-ExecutionPolicy` command returns `RemoteSigned`. If the execution policy is anything other than `RemoteSigned` you must use the `Set-ExecutionPolicy` command to switch to `RemoteSigned` before you can integration XenDesktop into Leostream.

## Creating a Citrix XenApp Services Site in XenDesktop 5

When working with XenDesktop 5.x, Leostream establishes HDX connections by connecting the user to their resources via a Citrix XenApp Services Site. Therefore, if you plan to use HDX in conjunction with Leostream, you must create a XenApp Services Site that includes your XenDesktop server farm, for example:

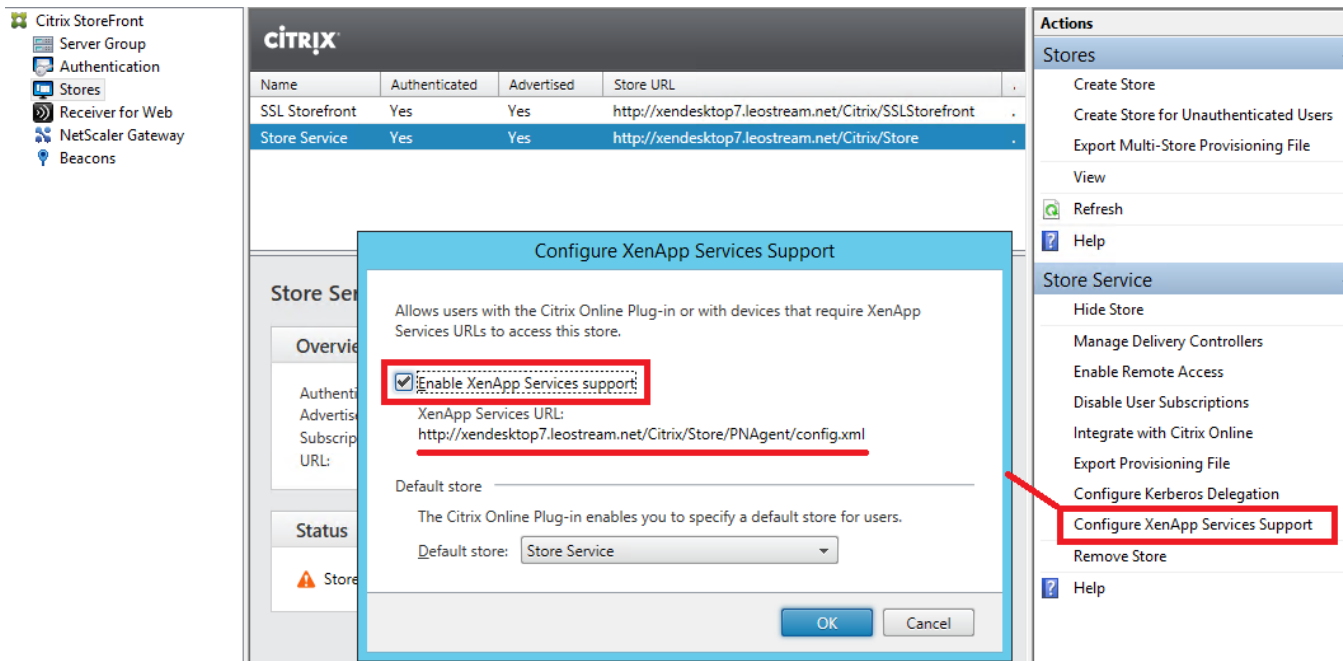


Ensure that you use the default **Prompt** authentication method for the site.

## Setting up a Citrix Storefront in XenDesktop 7

When working with XenDesktop 7, in order to establish an HDX connection to a desktop, Leostream pushes the desktop assignment into the Citrix Studio. After the assignment is pushed into the Citrix Studio, Leostream uses a Citrix XenApp Services Site to obtain the ICA-file required to establish the HDX connection.

You must create a Citrix Storefront that enables the XenApp Services Site, as shown, for example, in the following figure.



Make note of the XenApp Services URL underlined in the previous figure.

## Configuring Client Devices

You must install the Citrix Receiver and Leostream Connect client on each client device.

## HDX Connections to Persistent Desktops Assigned by Leostream


### Step 1: Create a Citrix XenDesktop Center

After the Leostream Agent is installed on the Citrix Studio, create the XenDesktop center, as follows.

1. Go to the **> Resources > Centers** page.
2. Click **Add Center**. The **Add Center** form opens.
3. Select **Citrix XenDesktop 5** or **Citrix XenDesktop 7** from the **Type** drop-down menu. The form updates, as follows:

The screenshot shows the 'Add Center' form with the following fields and annotations:

- Type:** Citrix XenDesktop 7 (dropdown menu)
- Name:** (empty text field) - Annotation: "Enter a display name for this center."
- XenDesktop Controller address:** (empty text field) - Annotation: "Enter the IP address of the primary Desktop Studio in your XenDesktop farm."
- Agent RPC port:** 8080 (text field) - Annotation: "You must install the Leostream Agent on the Desktop Studio. Enter the port number that the Leostream Agent listens on. The default port is 8080. To avoid conflicts, change the Leostream Agent port here and on the Leostream Agent Control Panel to, for example, 8730."
- Catalog for Leostream assignments:** Leostream Desktops (text field) - Annotation: "Specify the Catalog that will hold all desktops assigned by Leostream. Do not manually create this folder. The Connection Broker automatically builds the folder when you save the Center."
- Username:** (empty text field) - Annotation: "Enter the username and password for a user with administrator rights to the server running the Desktop Studio. The user name must include the user's domain, for example leostream\admin."
- Password:** (empty text field)
- Refresh interval:** 10 minutes (dropdown menu)
- Notes:** (empty text area)
- Buttons:** Save, Cancel

4. Enter a name for the center in the **Name** edit field.
  5. In the **XenDesktop Controller address** edit field, enter the address the Connection Broker uses to communicate with the Citrix Studio in your XenDesktop farm.
  6. In the **Agent RPC port** edit field, enter the port that the Leostream Agent installed on the Citrix Studio listens on. Ensure that this port is different from any of the ports used by the Studio.
  7. In the **Catalog for Leostream assignments** edit field, enter the name of the catalog you want to hold all desktops assigned by Leostream.
-  Do not manually create this catalog. The Connection Broker automatically creates a Static machine catalog with this name when you save the **Create Center** form. This catalog is used when assigning persistent desktops to users.
8. In the **Username** edit field, enter the username for a user that has administrator rights to the desktop where the Citrix Studio is installed. Include the user's domain in the field, in the form: `domain\username`.
  9. Enter this user's password in the **Password** edit field.
  10. Select a value from the **Refresh Interval** drop-down menu to indicate how often the Connection Broker checks if the XenDesktop center is still online.
  11. Click **Save**.

After you save the center, ensure that the center is listed as **Online** on the **> Resources > Centers** page and that your new catalog was created in your Citrix Studio.

## Step 2: Define an HDX Protocol Plan

You can establish HDX connections from Leostream Connect and the Leostream Web client, which use the **Leostream Connect and Thin Clients Writing to Leostream API** and **Web Browser** sections of protocol plans, respectively. The following instructions apply to both of sections.

To configure a protocol plan that establishes HDX connections to resources assigned by Leostream:

1. In the **Edit Protocol Plan** or **Create Protocol Plan** form, select **1** from the **Priority** drop-down menu associated with Citrix HDX, as shown in the following figure.

**Leostream Connect and Thin Clients Writing to Leostream API**

RDP and RemoteFX Priority: Do not use

Citrix HDX Priority: 1

Configure the following two fields to establish HDX connections to a desktop in a Leostream Pool

Create assignment in selected XenDesktop center

Select ...

Site URL for XenApp Services Site

Enter Site URL for XenApp Services Site that publishes connections for the selected XenDesktop center, for example: `http://192.168.0.1/Citrix/PNAgent`

Alternate port for remote viewer port check

8080

If policies use the remote viewer port check to invoke backup pools, enter the port to check for HDX connections. (For example, Citrix VDA port 8080. Do not enter 1494.)

2. From the **Create assignments in selected XenDesktop center** drop-down menu, select the XenDesktop center where Leostream will push new desktop assignments.

Ensure that this XenDesktop farm does not contain a Desktop Group that assigns the desktops to which this protocol plan will be applied. Citrix restricts you to placing a particular desktop in a single Desktop Group.

3. From the **Site URL for XenApp Services Site** field, enter the Site URL to the XenApp Services Site that publishes connections for the Citrix Studio select in the previous step.

In XenDesktop 5, this URL typically takes the following form:

```
http://192.168.226.133/Citrix/PNAgent
```

In XenDesktop 7, this URL typically takes the following form:

```
http://192.168.0.1/Citrix/StoreFront/PNAgent
```

4. In the **Alternate port for remote viewer port check** field specify the port number that the Connection Broker pings to determine if the desktop is available for a connection. Do not enter port 1494 as the Citrix Virtual Desktop Agent manages that port and the Connection Broker port check will always fail.
5. Ensure that no other protocol in the section select **1** for their priority.
6. Click **Save** to save the protocol plan.



In XenDesktop 5, to launch the HDX connection in fullscreen mode, add the following line in the application section of the default.ica file provided by XenDesktop, typically located in C:\inetpub\wwwroot\Citrix\PNAgent\conf\.

```
DesktopViewer-ForceFullScreenStartup=true
```

For more information, see the following Citrix Forum article:

<http://forums.citrix.com/thread.jspa?threadID=286952&start=0&tstart=0>

### Step 3: Create a Pool of HDX-Enabled Desktops

When working with Citrix XenDesktop, create a pool that contains only desktops that users will connect to using HDX. These desktops must be inventoried in Leostream from an Active Directory center (see “Active Directory Centers” in Chapter 5 of the Connection Broker Administrator’s Guide.)

When Leostream pushes an assignment into XenDesktop, the Desktop Group contains the name of the Leostream pool that contains the desktop. Because Citrix restricts desktops to be in a single Desktop Group, ensure that desktops appear in a single Leostream pool.

For more information on create pools, see “Chapter 7: Creating Desktop and Application Pools” in the Connection Broker Administrator’s Guide.

### Step 4: Use the HDX Protocol Plan in Policies

In the **Desktop Assignment from Pools** section of the **Edit Policy** or **Create Policy** form, associate the protocol plan you created in Step 2 with the pool created in Step 3, for example:

**Desktop Assignments from Pool "HDX VM"**

**When User Logs into Connection Broker**

Number of desktops to offer: 1

Pool: HDX VM

Backup pool: Select ...

Offer desktops from this pool: To all users of this policy

Select desktops to offer based on: User ("follow-me" mode)

Display desktop to user as: Pool name

Allow users to reset offered desktops: Not allowed

Offer running desktops: Yes, regardless of Leostream Agent status

Offer stopped and suspended desktops: No

Offer desktops with pending reboot job: Yes

Desktop selection preference: Favor desktops previously assigned to this user

**When User is Assigned to Desktop**

☐ Revert the desktop to its most-recent snapshot

☐ Confirm desktop power state

☐ Log out any rogue users

☐ Enable single sign-on to desktop console (VNC and PCoIP, only)

☐ Prevent user from manually releasing desktop

☐ Adjust time zone to match client (Leostream Connect and HP SAM, only)

**Plans**

Protocol: HDX

Power control: Default

Release: Default





The Citrix VDA locks the remote session when you disconnect from an HDX session, causing the Leostream Agent to send a lock event to the Connection Broker. Because a disconnect event does not occur, you cannot use the **When user disconnects** section of power control or release plans with HDX connections.

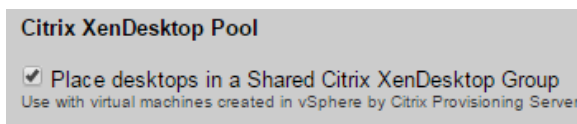
For more information on how Leostream pushes the assignment into Citrix XenDesktop, see the [Leostream Quick Start Guide for Citrix XenDesktop 7](#)

## HDX Connections to Non-Persistent Desktops Assigned by Leostream

A non-persistent desktop is a virtual machine with an operating system that is streamed down from Citrix Provisioning Services. Leostream can manage HDX connections to non-persistent desktops that are hosted in VMware vSphere and inventoried in the Connection Broker using a vCenter Server center.

The procedure for managing HDX connections to non-persistent desktops is identical to managing persistent desktops, with the following two exceptions.

1. When creating a pool of non-persistent desktops in step 3, ensure that you select the **Place desktops in a Shared Citrix XenDesktop Group** option at the bottom of the **Create Pool** form, for example:



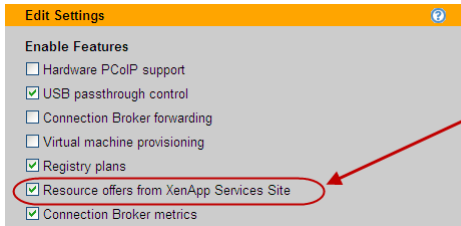
2. Use Leostream power control plans to instruct Leostream to reboot the virtual machine at the appropriate time, which then restreams the operating system. Ensure that you use this power plan in the policy created in step 4.

## HDX Connections to Resources Assigned by Citrix

In this scenario, Leostream *pulls* desktop assignments from a XenApp Services Site. The Connection Broker then always uses the Citrix Receiver and HDX to connect the user to a resource offered from a Citrix XenApp Services Site.

### Step 1: Enable the Citrix XenApp Services Site Feature

You must specifically enable the feature to offer resources from a Citrix XenApp Services Site by selecting the **Resource offers from XenApp Services Site** option on the **> System > Settings** page, as shown in the following figure.



See [Creating a Citrix XenApp Services Site in XenDesktop 5](#) or [Setting up a Citrix Storefront in XenDesktop 7](#) for information on setting up a XenApp Services site to use with Leostream.

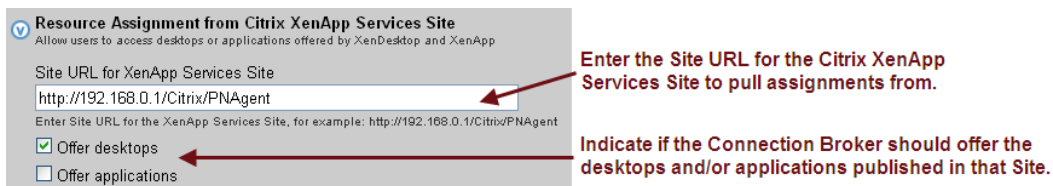
## Step 2: Configure the Policy

To configure a policy to offer the user Citrix XenDesktop resources:

1. In the **Edit Policy** or **Create Policy** form, scroll down to the **Desktop Assignment from Citrix XenApp Services Site** section.
2. In the **Site URL for XenApp Services Site** enter the URL for the XenApp Services Site, for example, in XenDesktop 5:

`http://xenapp_services_site.yourcompany.com/Citrix/PNAgent`

as shown in the following figure



3. When a user with this policy logs into the Leostream Connection Broker, Leostream simulates a log in to the specified Citrix XenApp Services Site to determine which desktops and applications are assigned by XenDesktop and XenApp. Use the **Offer desktops** and **Offer applications** check boxes to indicate which of these resources Leostream should offer to the user.

## Citrix ICA

The Connection Broker uses Citrix ICA to connect to any application or desktop published in a Citrix XenApp farm. The Connection Broker can launch Citrix ICA connections from Leostream Connect (the Windows or Java version) and the Leostream Web client, as well as from any Wyse WTOS thin client.

In order to launch ICA connections from a desktop, laptop, or thin client running Leostream Connect, you must install the Citrix Online Plugin on the client device. The Leostream Web client supports the Citrix Client for Java.

The ICA-files used when connecting to Citrix XenApp applications and desktops are set in the **Citrix XenApp**

(ICA) Configuration section of the protocol plan, shown in the following figure.



## Using the Citrix Online Plugin

The configuration files in the **Citrix Plugin** subsection of the protocol plan configure how the ICA session is established when using the Citrix Online Plugin. These ICA-files override any ICA-file locally assigned to the user.

The following default configuration file for applications uses seamless windows and provides single sign-on by passing the user's password in plain text.

```
[Encoding]
InputEncoding=ISO8859_1

[WFClient]
Version=2
TcpBrowserAddress={IP}
HttpBrowserAddress={IP}

[ApplicationServers]
{CITRIX_RESOURCE}=

[{CITRIX_RESOURCE}]
Address={IP}
BrowserProtocol=HTTPOntTCP
Password={SCRAMBLED_PASSWORD}
ClientAudio=Off
DesiredColor=4
DesiredWinType=8
Domain={DOMAIN}
InitialProgram="#"{CITRIX_RESOURCE}"
ScreenPercent=0
TWIMode=On
TransportDriver=TCP/IP
UseDefaultWinSize=Off
Username={USER}
WinStationDriver=ICA 3.0
```

The following default configuration file for desktops also provides single sign-on for desktops by passing the user's plain password:

```
[Encoding]
InputEncoding=ISO8859_1

[WFClient]
Version=2
TcpBrowserAddress={IP}
HttpBrowserAddress={IP}

[ApplicationServers]
{CITRIX_RESOURCE}=

[{CITRIX_RESOURCE}]
Address={IP}
BrowserProtocol=HTTPonTCP
Password={SCRAMBLED_PASSWORD}
ClientAudio=Off
DesiredColor=4
DesiredWinType=0
Domain={DOMAIN}
InitialProgram="#"{CITRIX_RESOURCE}"
ScreenPercent=0
TWIMode=Off
TransportDriver=TCP/IP
UseDefaultWinSize=Off
Username={USER}
WinStationDriver=ICA 3.0
```

In these files, the Connection Broker replaces the {IP} dynamic tag with the IP address or hostname of the XenApp center that publishes this resource, and the dynamic tag {CITRIX\_RESOURCE} with the name of the desktop or application being launched.

Connection Broker version 7.0.52 and higher replaces the {SCRAMBLED\_PASSWORD} dynamic tag with the user's password scrambled to prevent casual eavesdropping. To pass a plain password, change the Password ICA-file parameter to ClearPassword and the {SCRAMBLED\_PASSWORD} dynamic tag to {PLAIN\_PASSWORD}



When launching Citrix applications from the Java version of Leostream Connect, ensure that the XenApp center in the Connection Broker was created using the IP address of the Citrix farm, not the fully qualified domain name.

## Launching Desktop Connections in Fullscreen

To launch a desktop in full screen, add the following lines to the **Desktop configuration file**:

```
UseFullScreen=Yes
NoWindowManager=True
```

And remove the following lines to the **Desktop configuration file**:

```
UseDefaultWinSize=Off
ScreenPercent=0
```

See the Citrix Knowledge Center article [CTX14753](#) to access documentation that describes the ICA-file parameters. A complete list of parameters is given in the Parameters chapter of the Citrix document [ini file reference.pdf](#).

## Using the Citrix Client for Java

When launching a XenApp resource from the Leostream Web client, you can choose to use either an installed Citrix Online Plugin or download and run the Citrix Client for Java. Select the **Use the Citrix Client for Java when connecting from a Web browser** option to instruct the Connection Broker to use the Citrix Client for Java to launch the XenApp resources. The Citrix Client for Java is a Java applet, which is downloaded and run when the user launches one of their applications. When using the Citrix Client for Java, no additional software needs to be installed on the client device.



Ensure that the appropriate Java version is available in your Web browser when using the Citrix Client for Java. Consult your Citrix documentation for Java version requirements.

The **Application configuration file** and **Desktop configuration file** edit fields contain the code that runs the Java applet. To use the JICA client with Safari Web browsers, wrap the applet code in the protocol plan in the appropriate HTML tags, for example:

```
<html>
  <head>
    <title>Connection Broker Title</title>
  </head>
  <body>
    <applet name="javaclient"
      code="com.citrix.JICA"
      codebase="java/Citrix"
      archive="JICAEngN.jar"
      width="640"
      height="480">
      <param name="Username" value="{USER}">
      <param name="Domain" value="{DOMAIN}">
      <param name="Password" value="{PLAIN_PASSWORD}">
      <param name="HTTPBrowserAddress" value="{IP}">
      <param name="Address" value="{CITRIX_RESOURCE}">
      <param name="InitialProgram" value="#{CITRIX_RESOURCE}">
      <param name="End" value="index.pl">
      <param name="cabinets" value="JICAEngM.cab">
    </applet>
  </body>
</html>
```

## Uploading New Client Versions

The Leostream Connection Broker includes a version of the Citrix Java client, which is located in the Broker's `java/Citrix` directory. You can upload new versions of the Citrix Java client using the **Install third-party content** option on the **> System > Maintenance** page. See the "Installing and Removing Third Party Content" section in the [Connection Broker Administrator's Guide](#) for complete instructions on uploading a new client.

After you upload a new version of the Citrix Java client, modify the `codebase` line in the applet code, as follows:

```
codebase="tpc"
```

## Launching the Client in a new Window

If this protocol plan is used in a policy that selects the **Launch web client connections in new window** option, you can use the **Parameters for connections opened in new window** edit field to configure the appearance of the new window. The Connection Broker uses the JavaScript `window.open` function to launch the new window. For a list of parameters, see:

[http://www.w3schools.com/jsref/met\\_win\\_open.asp](http://www.w3schools.com/jsref/met_win_open.asp)

Enter parameters as a comma-separated list, for example:

```
left=0,height=500,width=700,toolbar=1,status=1
```

If your policy launches the Java applet in a new window, you can modify the applet code in the protocol plan to close the new window after the user disconnects from their application. Change the `End` parameter from `index.pl` to `javascript:window.close();`, for example:

```
<param name="End" value="javascript:window.close();">
```

## Ericom® Blaze

Ericom **Blaze** provides software acceleration for Microsoft RDP connections. Blaze is available when connecting from the Windows and Java versions of Leostream Connect.

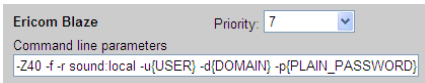
To use Blaze, you must install the Blaze client on the desktop running Leostream Connect, and install the Blaze Server on the remote desktop.



When installing the Blaze client, ensure that you do *not* select the option to automatically associated RDP files with Blaze. If you do select this option, Blaze intercepts all RDP files and user's logging into a desktop with a protocol plan that prefers native RDP with automatically switch to Blaze.

The following figure shows the section of the protocol plan associated with Blaze. The Connection Broker

supports only command line parameters when launching Blaze. You cannot specify a configuration file.



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

Use the **Command line parameters** field to customize the Blaze connection. Blaze supports the following command line parameters.

- `-u`: Specifies the user name
- `-d`: Specifies the user's domain
- `-p`: Specifies the user's password
- `-s`: Run in a shell
- `-c`: Indicates the working directory
- `-g`: Sets the desktop geometry (width x height)
- `-f`: Indicates the connection should open in full-screen mode
- `-G`: Shows connection bar
- `-U`: Open as a unified desktop that combines the remote and local desktops. Your desktop displays two Start menus and two task bars.
- `-M#`: Indicates which monitor contains open the remote session. Replace # with the monitor to use:
  - 1 = first monitor
  - 2=second monitor
  - -1 = primary monitor
  - -2 = secondary monitor
  - 0 = span all monitors
- `-b`: Force bitmap updates
- `-A`: Enable Seamless RDP mode
- `-K`: Keep window manager key bindings
- `-H[never|always|fullscreen]`: Use keyboard hook (for special Windows keys), default = fullscreen.
- `-T`: Window title
- `-a#`: Connection color depth. Defaults to `-a24` for XP and 2003, otherwise defaults to `-a32`.
- `-z`: Enable RDP compression. Do not use this parameter together with `-Z`.
- `-Z#`: Blaze image quality. Specify a number from 10 to 100) Do not use this parameter together with `-z`.
- `-x`: Provide an RDP5 experience (m[odem 28.8], b[roadband], l[an] or hex nr.)
- `-r`: Enable specified device redirection (this flag can be repeated)

#### For Linux only

- `-r comport:COM1=/dev/ttyS0`: Enable serial redirection of /dev/ttyS0 to COM1
- `-r comport:COM1=/dev/ttyS0, COM2=/dev/ttyS1`: Enable serial redirection of

- `/dev/ttyS0 to COM1, and /dev/ttyS1 to COM2`
- o `-r disk:floppy=/mnt/floppy`: Enable redirection of `/mnt/floppy` to floppy share
- o `-r disk:floppy=/mnt/floppy, cdrom=/mnt/cdrom`: Enable redirection of `/mnt/floppy` and `/mnt/cdrom`
- o `-r clientname=<client name>`: Set the client name displayed for redirected disks
- o `-r lptport:LPT1=/dev/lp0`: Enable parallel redirection of `/dev/lp0` to LPT1
- o `-r lptport:LPT1=/dev/lp0, LPT2=/dev/lp1`: Enable parallel redirection of `/dev/lp0` to LPT1 and `/dev/lp1` to LPT2
- o `-r printer:mydeskjet`: Enable printer redirection
- o `-r printer:mydeskjet=\"HP LaserJet IIIP\"`: Enter server driver as well

For all platforms

- o `-r sound:[local[:driver[:device]]|off|remote]`: Enable sound redirection, remote would leave sound on server
- o `-r clipboard:[off|CLIPBOARD]`: Enable clipboard redirection.

## 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](#).

The Leostream clients and agents that support Exceed onDemand include:

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



The Leostream Agent for Windows does not track user connections to an Exceed onDemand server running on a Windows server.

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

1. Go to the **> Plans > Protocol** 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 and **ActiveX 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 running on the Linux operating systems 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



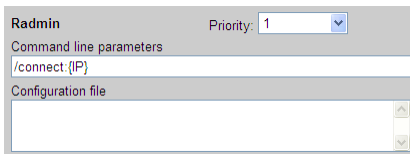
If a user logs out of, for example, a gnome desktop session but leaves the Exceed connection open, the Connection Broker does not receive a logout event from the Leostream Agent. The user must close the Exceed session to register a logout event

- 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.

## Famatech Radmin® 2.2 and 3.x Remote Viewer

You can use the Famatech Remote Administrator ([Radmin®](#)) software when connecting from the Windows or Java versions of Leostream Connect. To use Radmin, you must install the Radmin Viewer on the desktop running Leostream Connect, and install the Radmin Server on the remote desktop. When running the Radmin viewer on a Linux client, you must use **Wine** to run the Windows application on the Linux operating system.

The following figure shows the section of the protocol plan associated with the Radmin viewer.



Use the **Command line parameters** field to customize the Radmin viewer. The default command line syntax:

```
/connect{IP}
```

Where the Connection Broker replaces the dynamic tag {IP} with the hostname or IP address of the remote desktop. If the remote desktop is not using the default Radmin port of 4899, use the following syntax:

```
/connect{IP}:nnnn
```

Where *nnnn* is the port used by the Radmin server.

The following list describes a subset of other available command line parameters. See the [Radmin User Manual](#) for a complete list of available command line parameters.

- `/noinput`: Specify a view-only connection mode view of the remote screen
- `/fullscreen`: Specify the full screen view mode
- `/updates:nn`: Specify a maximum number of screen updates per second



When using Radmin with the Windows version of Leostream Connect, you will be prompted for the Radmin installation folder before launching a remote session.

## 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 5.2, 5.3, 5.4, 6.x, and 7.0.1. You must use RGS version 6.0.2 or higher to use HP Velocity and other new optimization features available with RGS version 6.



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:

```
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.

## 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 `IsMatchReceiverResolutionEnabled` and `IsMatchReceiverPhysicalDisplaysEnabled` 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

RGS 6.0 introduced HP Velocity and Advanced Video Compression features to improve RGS performance over WAN connections. You can utilize these new features when establishing RGS connections using Leostream Connect when using RGS version 6.0.2 or higher. 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 6.0 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 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:
  - 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.
  - 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.
  - 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

## Remembering Window Position for HP RGS Connections

When running on a client with multiple monitors, the Java implementation of Leostream Connect can track the window position for HP RGS remote sessions. By tracking the window position, Leostream Connect can reopen the remote session for a particular desktop on the same physical display every time the user launches the connection.

To enable window position tracking for a particular client, add the following line to the `lc.conf` file on the client.

```
enable_window_tracking = true
```

Then, to turn on window tracking for a particular desktop and user:

1. Create a protocol plan that contains the following lines in the `configuration file` field associated with HP RGS.

```
Rgreceiver.Session.{SESSION}.VirtualDisplay.IsPreferredResolutionEnabled=1
Rgreceiver.Session.{SESSION}.RemoteDisplayWindow.X={VALUE:x}
Rgreceiver.Session.{SESSION}.RemoteDisplayWindow.Y={VALUE:y}
```

See the [HP Remote Graphics Software User Guide](#) for more information on these `Rgreceiver` parameters.

{SESSION} is a Leostream dynamic tag. Leostream Connect automatically adjusts the value for {SESSION} when the user connects to multiple desktops using HP RGS. {VALUE:x} and {VALUE:y} are additional dynamic tags that Leostream Connect uses to label and store the X and Y position of the remote session window for the desktop with session ID {SESSION}.

2. Assign this protocol plan to the pool or pools in the user's policy, as shown in the following figure.

Create the Protocol Plan with the appropriate RGS receiver parameters and then assign this protocol plan to the appropriate pool or pools of desktops in the user's policy.

**Create Protocol Plan**

Plan name: Track RGS Windows

Leostream Connect and Thin Clients Writing to Leostream API

RDP Priority: 2

Command line parameters:

Configuration file:

```
screen mode id:i:1:2
desktopwidth:i:1024
desktopheight:i:1768
```

RGS Priority: 1

Configuration file:

```
Rgreceiver.Session.
{SESSION}.VirtualDisplay.IsPreferredResolutionE
nabled=1
Rgreceiver.Session.
{SESSION}.RemoteDisplayWindow.X={VALUE:x}
Rgreceiver.Session.
{SESSION}.RemoteDisplayWindow.Y={VALUE:y}
```

**Create Policy**

General Policy Properties

Policy name: RGS

☐ Auto-launch remote viewer session if only one desktop is offered (Web client, only)

Maximum number of desktops assigned: <No Limit>

Maximum number of desktops that can be assigned across all Desktop pools. Does not apply to applications or desktops offered from the Application Pool

Expire user's session: Never

Desktop Assignment from Pools

Pool: Select ...

When User Logs into Connection Broker

Number of desktops to offer: 1

Select desktops to offer based on: User ("follow-me" mode)

Display to user as: Desktop name

Allow users to reset offered desktops: Not allowed

☐ Offer running desktops without a Leostream Agent

☐ Offer stopped and suspended desktops

When User is Assigned to Desktop

☐ Revert the desktop to its most-recent snapshot

☐ Log out any rogue users

☐ Enable single sign-on to desktop console (VNC and PCoIP, only)

☐ Prevent user from manually releasing desktop

☐ Adjust time zone to match client (Leostream Connect and HP SAM, only)

Plans

Protocol: Track RGS Windows

## Single Sign-On with HP RGS

To achieve single sign-on with HP RGS:

- When installing the RGS Sender on a Windows desktop, ensure that the HP RGS Easy Login option is configured. With the Easy Login option enabled, you do not need to install the Leostream Agent on the desktop. If you do not want to use the Easy Login option, install the Leostream Agent on the remote desktop with the single sign-on task enabled.
- 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.

## 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

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.

**Edit Protocol Plan**

Plan name: RGS

▼ Leostream Connect and Thin Clients Writing to Leostream API

RDP and RemoteFX Priority: Do not use ▼

ThinAnywhere Priority: Do not use ▼

VMware View Priority: Do not use ▼

Citrix HDX Priority: Do not use ▼

HP RGS Priority: 1 ▼

Send user login name as: {USER}


Configuration file:  
 Rgreceiver.IsBordersEnabled=1  
 Rgreceiver.IsBordersEnabled.IsMutable=0  
 Rgreceiver.Network.HPVelocity.Mode=1

☒ Allow user to modify RGS configuration file parameters

Parameter	Default value	Default custom value	Display values
<input type="checkbox"/> Image quality	65 ▼		Edit
<input type="checkbox"/> Match resolution	No ▼		Edit
<input type="checkbox"/> Match layout	No ▼		Edit
<input type="checkbox"/> Resolution	1024x768 ▼		Edit
<input type="checkbox"/> Show borders	No ▼		Edit

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 which 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> for more information on the proper syntax for configuring RGreceiver parameters.

Parameter	RGreceiver Parameter in the Protocol Plan	Leostream Dynamic Tag
Show borders	.IsBordersEnabled	{BORDERS}
Match resolution	.IsMatchReceiverResolutionEnabled	{MATCH_RESOLUTION}
Match layout	.IsMatchReceiverPhysicalDisplaysEnabled	{MATCH_DISPLAYS}
Resolution	.Session.{SESSION}.VirtualDisplay.IsPreferredResolutionEnabled .Session.{SESSION}.VirtualDisplay.PreferredResolutionHeight .Session.{SESSION}.VirtualDisplay.PreferredResolutionWidth	{RESOLUTION_ENABLED} {RESOLUTION_HEIGHT} {RESOLUTION_WIDTH}
Image quality	.ImageCodec.Quality	{IMAGE_QUALITY}

For example, if you allow the user to configure all available parameters, you must add the following lines to your configuration file.

```
Rgreceiver.IsBordersEnabled={BORDERS}
Rgreceiver.ImageCodec.Quality={IMAGE_QUALITY}
Rgreceiver.IsMatchReceiverResolutionEnabled={MATCH_RESOLUTION}
Rgreceiver.IsMatchReceiverPhysicalDisplaysEnabled={MATCH_DISPLAYS}
Rgreceiver.Session.{SESSION}.VirtualDisplay.IsPreferredResolutionEnabled={RESOLUTION_ENABLED}
Rgreceiver.Session.{SESSION}.VirtualDisplay.PreferredResolutionHeight={RESOLUTION_HEIGHT}
Rgreceiver.Session.{SESSION}.VirtualDisplay.PreferredResolutionWidth={RESOLUTION_WIDTH}
```

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.

- From the **Default value** drop-down menus, indicate the value to use if the user has not customized the parameter.
- 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 *heightxwidth* where *height* and *width* are in pixels and there is no space between the numbers and the x.
- The drop-down menus in the end-user dialog displays 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:
  - Click the **Edit** link in the **Display value** column
  - In the **Edit Display Values** form that opens, enter user-friendly names into the **Display value** edit field for each possible internal value.
  - 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.

## 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.

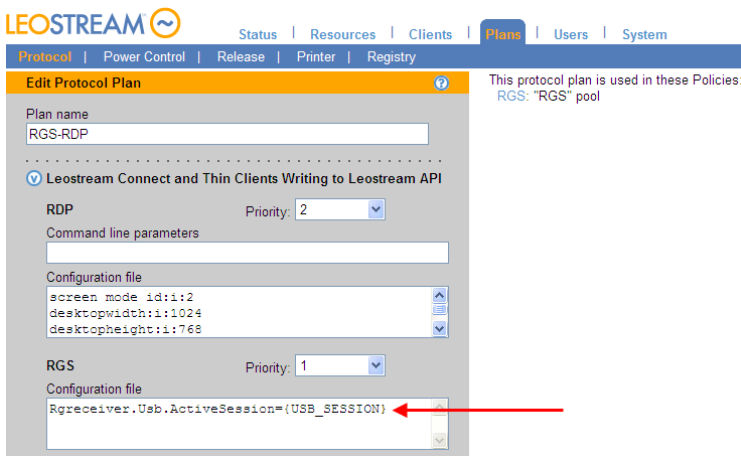
To turn on the sidebar for USB access:


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

```
sidebar_enabled = true
```

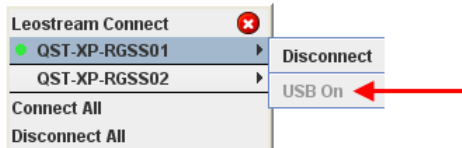
2. In the protocol plan assigned to users that connect to desktops using HP RGS, add the following line to the **Configuration file** field for HP RGS, as shown in the following figure.

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



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

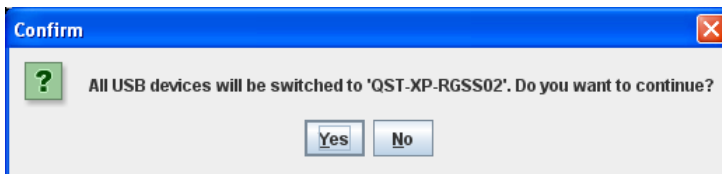
When a user logs in through Leostream Connect, by default, the first desktop they connect to using HP RGS has access to all USB devices. The sidebar menu for this desktop, shown in the following figure, displays a **USB On** menu item.



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.



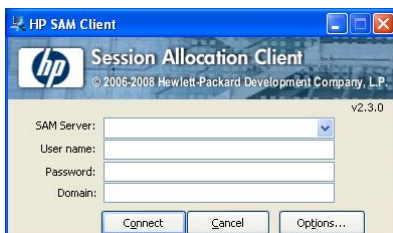
HP RGS simultaneously allows access to USB devices from a single desktop.



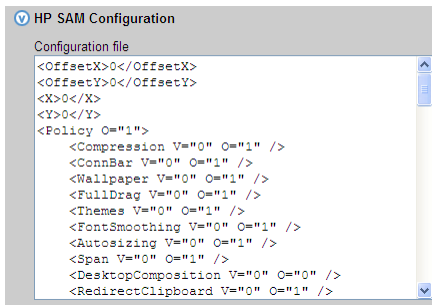
If you disconnect from the RGS session that has access to USB devices, Leostream Connect automatically switches all USB devices to the next active RGS session.

## 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, shown in the following figure.



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:

```
<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.

```
<DynamicPolicy>Rgreceiver.IsMatchReceiverResolutionEnabled =value</DynamicPolicy>
```

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.

```
<DynamicPolicy>Rgreceiver.IsMatchReceiverResolutionEnabled.IsMutable=0</DynamicPolicy>
```

### 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:

```
<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.

```
<DynamicPolicy>Rgreceiver.IsMatchReceiverResolutionEnabled.IsMutable=0</DynamicPolicy>
<DynamicPolicy>Rgreceiver.IsMatchReceiverResolutionEnabled=1</DynamicPolicy>
<DynamicPolicy>Rgreceiver.IsMatchReceiverPhysicalDisplaysEnabled.IsMutable=0</DynamicPolicy>
<DynamicPolicy>Rgreceiver.IsMatchReceiverPhysicalDisplaysEnabled=1</DynamicPolicy>
```

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

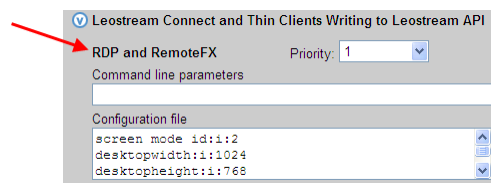
```
<DynamicPolicy>Rgreceiver.IsBordersEnabled=0</DynamicPolicy>
```

### 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, shown in the following figure, 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:

```
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.

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.

## 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.

<http://windowshelp.microsoft.com/Windows/en-US/help/142d58b8-43f0-432f-93bb-7653333905911033.mspx>

**/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. For an online description of the parameters, see the following links:

[http://technet.microsoft.com/en-us/library/ff393699\(v=ws.10\).aspx](http://technet.microsoft.com/en-us/library/ff393699(v=ws.10).aspx)

***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.

Value	Setting
0	Use a single monitor
1	Use all monitors

***span monitors:i***

Indicates if the remote session should be spanned across multiple monitors.

Value	Setting
0	Spanning is off
1	Spanning is on

***screen mode id:i***

Determines if the remote session is opened in a window or in full screen.

Value	Setting
2	Open in full screen
1	Open in a window

***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:i***

Corresponds to the desktop height (in pixels) on the **Display** tab in Remote Desktop Connection **Options** dialog.

For ActiveX remote viewers, the variables are:

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



If you specify a screen width greater than the RDP maximum (1600 pixels) you receive an error message. This may occur if you specify a full screen size and have a large screen.

***connection type:i***

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:i***

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:s***

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:

<http://msdn.microsoft.com/en-us/library/ms632612.aspx>

***auto connect:i:0***

This setting is not used by desktop computers or by Windows CE-based clients.

***Full Screen Title***

For ActiveX remote clients, the variable is:

```
MsRdpClient.FullScreenTitle =
```

***full address:s {IP}***

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:

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

***password 51:b: {RDP\_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. By 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. If you pass the `{RDP_PASSWORD}` tag to one of these client devices, your users will not single sign-on to their desktops. 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

Value	Setting
0	Compression is off
1	Compression is on

***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.

Value	Setting
0	On the local computer
1	On the remote computer
2	In full-screen mode only

***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.

Value	Setting
0	Bring to this computer
1	Leave at remote computer
2	Do not play



***redirectclipboard:i:i***

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.

Value	Setting
0	Clipboard is not enabled
1	Clipboard is enabled

***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.

Value	Setting
0	Drives are not automatically reconnected
1	All drives are automatically reconnected

For ActiveX remote viewers, the variable is:

```
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.

Value	Setting
{DRIVE:DVD}	All DVD drives are automatically connected. No other drives are connected.
{DRIVE:CD}	All CD drives are automatically connected. No other drives are connected

***devicestoredirect:s***

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:i:0***

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.

Value	Setting
0	Microsoft Point of Service for .NET (POS for.NET) device redirection is disabled
1	Microsoft Point of Service for .NET (POS for .NET) device redirection is enabled

***redirectprinters:i***

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.

Value	Setting
0	Printers are not automatically reconnected
1	Printers are automatically reconnected

For ActiveX remote viewers, the variable is:

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

***redirectcomports:i***

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.

Value	Setting
0	COM ports are not automatically reconnected
1	COM ports are automatically reconnected

***redirectsmartcards:i***

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.

Value	Setting
0	Smart cards are not automatically reconnected
1	Smart cards are automatically reconnected

For ActiveX remote viewers, the variable is:

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

***displayconnectionbar:i***

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.

Value	Setting
0	Connection bar does not appear
1	Connection bar appear

For ActiveX remote viewers, the variable is:

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

***username:s; {USER}***

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:

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

***domain:s {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:

```
MsRdpClient.Domain = "{DOMAIN}"
```

***alternate shell:s***

Determines 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:s***

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:i***

Determines 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.

Value	Setting
0	Wallpaper appears
1	Wallpaper does not appear

***disable full window drag:i***

Determines 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.

Value	Setting
0	Folder contents appear while dragging
1	Folder contents do not appear while dragging

***disable menu anims:i***

Determines 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.

Value	Setting
0	Menu and window animations are permitted
1	Menu and window animations are not permitted

***disable themes:i***

Determines 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.

Value	Setting
0	Themes are permitted
1	Themes are not permitted

***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.

Value	Setting
0	Caching is not enabled
1	Caching is enabled

## NoMachine NX

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

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 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 NX version 4.0. To support older versions of NX, launch the NX 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>

NX 3.5, including the NX Client package, is no longer available for download from the NoMachine website. All registered customers with valid subscriptions can continue to use and download version 3.5.0 from their designated customer area.

## Launching NX Connections from the Web Client

Users that log in using the Leostream Web client can launch NX connections using either the native NX client or the NX Web Companion. Settings in the user's protocol plan determine which client is used. To setup a protocol plan for the Web client:

1. Go to the **> Plans > Protocol** 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 **Select NX client type for connection** drop-down menu in the **NoMachine NX** section, shown in the following figure, to determine which client to use when the user logs in from the Leostream Web client.

The screenshot shows the 'Web Browser' configuration panel. Under the 'NoMachine NX' section, the 'Select NX client type for connection' dropdown is open, showing 'Native client' as the selected option. A red arrow points to this dropdown with the text: 'Select which NX client to launch when users log into the Leostream Web client.'

- Select **Native client** to use the natively installed NX client. In this case, the Leostream Web client downloads an NXS-file that defines the desktop connection to establish. Your Web browser must be configured to associate NXS-files with the NX client, and to automatically launch the NX client upon downloading the file.
- Select **Java client** to use the Java version of the NX client. In this case, the Leostream Web client installs the NX Java applet and automatically launches the desktop connection.

The Connection Broker includes the NX Java applet for NoMachine version 3.5. To connect to NoMachine 4.0, you must upload a newer version of the applet into the Connection Broker (see [Upgrading the NX Java Applet](#)).

5. Use the **Configuration file** field to define NX parameters for the connection. This field applies to the native NX client and NX Web Companion.

## NX Configuration File

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

Option Key	Purpose
User	(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.
Auth	(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 <b>configuration file</b> 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.
Server host	(Default = {IP}) The Connection Broker replaces the dynamic tag {IP} with the hostname or IP address of the NoMachine server.
Public Key	<p>Enter the key into the configuration file by placing an option key <code>Public Key</code> after the <code>Login Method</code> option, as follows.</p> <pre> ... &lt;option key="Login Method" value="nx" /&gt; &lt;option key="Public Key" value="" -----BEGIN DSA PRIVATE KEY----- &lt; Insert DSA Key here&gt; -----END DSA PRIVATE KEY----- /&gt; ... </pre>

## Upgrading the NX Java Applet

If you are connecting to Linux machines running NoMachine version 4, you can download a compatible NX Web Companion here:

<https://www.nomachine.com/download/download&id=73>

To install the new Web Companion in your Connection Broker, use the Connection Broker virtual machine console to enable SSH access to your Connection Broker. See the [Connection Broker Virtual Appliance Guide](#) for complete instructions.

After you enable SSH, copy the Web Companion RPM file into your Connection Broker and install the file using the following command:

```
sudo rpm -i nomachine-plugin_5.0.43_1_i686.rpm
```

The files unpack into the `/usr/NX/share` directory.

After unpacking the files, move the complete file set into the `/home/leo/app/plugin` directory.

## Setting User-Configurable NX Parameters

The configuration file in the NoMachine NX section of the protocol plan defines how the user's NX 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 configuration file parameters** option in the protocol plan. The form expands to include the additional fields shown in the following figure.

**NoMachine NX** Priority: 1

Configuration file

```
<!DOCTYPE NXClientSettings>
<NXClientSettings application="nxclient"
version="1.3" >
```

☒ Allow user to modify NX configuration file parameters

Parameter	Default value	Default custom value	Display values
<input type="checkbox"/> Connection type	LAN		Edit
<input type="checkbox"/> Desktop option	Virtual desktop		Edit
<input type="checkbox"/> Disable backingstore	True		Edit
<input type="checkbox"/> Resolution	Custom		Edit
<input type="checkbox"/> Span monitors	No		Edit
<input type="checkbox"/> Window manager	GNOME		Edit

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 which 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.



Parameter	Option Key	Original Text	Replace with
Connection type	Link speed	value="lan"	value="{CONNECTION_TYPE}"
Desktop option	Virtual desktop	value="false"	value="{VIRTUAL_DESKTOP}"
Disable backingstore	Disable backingstore	value="false"	value="{BACKINGSTORE}"
Resolution	Resolution Resolution height Resolution width	value="available" value="600" value="800"	value="{RESOLUTION}" value="{RESOLUTION_HEIGHT}" value="{RESOLUTION_WIDTH}"
Span Monitors	Spread over monitors	value="false"	value="{SPAN_MONITORS}"
Window manager	Desktop	value="kde"	value="{WM_TYPE}"

- From the **Default value** drop-down menus, indicate the value to use before the user customizes the parameter.
- 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.
- 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:
  - Click the **Edit** link in the **Display value** column
  - In the **Edit Display Values** form that opens, enter user-friendly names into the **Display value** edit field for each possible internal value.
  - 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.
- 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.

## Session Shadowing and Collaboration

The Connection Broker allows users that connect to desktops using the NoMachine NX protocol to collaborate by sharing their session or shadowing another user's session.

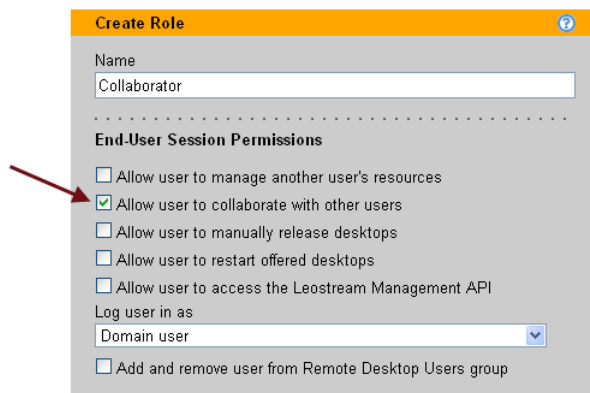


Collaboration is supported only for users that log in using the Leostream Web client.

### 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 NX sessions support collaboration.

1. **Building pools:** To simplify creating policies, construct pools that contain only desktops that support collaboration.
2. **Configuring roles:** You must explicitly give permission to each user that is either going to share their NX session or to shadow 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. **Configuring policies:** In addition to giving the user permission to collaborate with other users, you must indicate which NX sessions support collaboration. Use settings in the user's policies to indicate which sessions support collaboration, as follows.
  - a. Go to the **> Users > 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 3, select the **Enable session shadowing** option, as shown in the following figure.

**Desktop Assignments from Pool "kdg-NX"**

**When User Logs into Connection Broker**

Number of desktops to offer: <All>

Pool: kdg-NX

Offer desktops from this pool: To all users of this policy

Select desktops to offer based on: User ("follow-me" mode)

Display desktop to user as: Desktop name

Allow users to reset offered desktops: Not allowed

Offer running desktops: Yes, only if Leostream Agent is running

Offer stopped and suspended desktops: No

Offer desktops with pending reboot job: Yes

Desktop selection preference: Favor desktops previously assigned to this user

**When User is Assigned to Desktop**

☐ Revert the desktop to its most-recent snapshot

☐ Confirm desktop power state

☐ Log out any rogue users

☐ Enable single sign-on to desktop console (VNC and PCoIP, only)

☐ Prevent user from manually releasing desktop

☐ Adjust time zone to match client (Leostream Connect and HP SAM, only)

☒ Enable session shadowing (NoMachine NX only)

☐ View only shadowing, not interactive (NoMachine NX only)

**Plans**

Protocol: NX

- e. 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.
  - f. From the **Protocol** drop-down menu, select a plan that gives NoMachine NX the highest priority.
4. **Defining assignments:** After configuring a role and policy that support collaboration, you must configure the tables on the **> Users > Assignments** pages to assign that role and policy to the appropriate users. For example, the following figure gives all members of the Development group the role and policy that support collaboration.

**Edit Assignments for "DEV"**

Domain Name  
DEV

**Assigning User Role and Policy**  
In this section you can set up rules to assign Users to Roles and Policies based on their group membership. Optionally use the Order column to re-order the rows.

Attribute: memberOf Conditional: Exactly matches

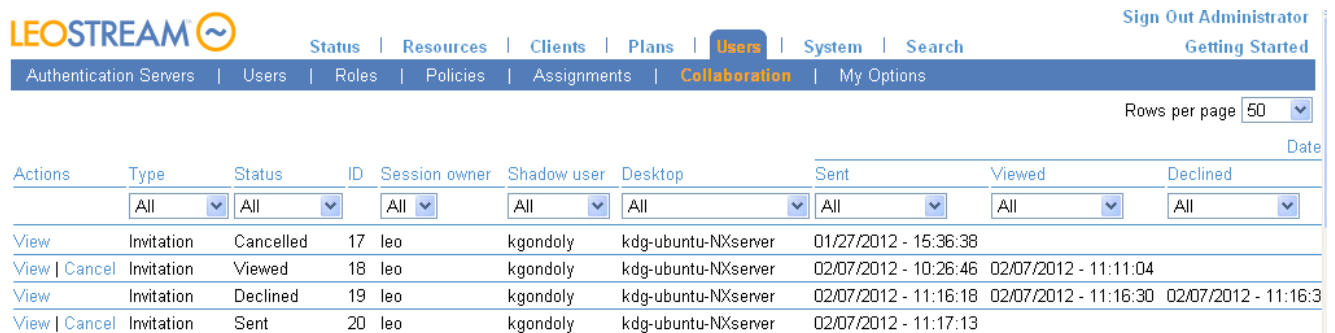
The Conditional setting controls how the user's Active Directory Attribute and entered Attribute Value must match, in order for the user to be assigned that role and policy.

Order	Attribute Value	Client Location	User Role	User Policy
1	Development	All	Collaborator	NX Shadowing

See "Using NoMachine NX Collaboration" in the [Connection Broker Administrator's Guide](#) for information on how to use the Leostream Web client to establish and connect to shadowed sessions.

## Managing Shadowed Sessions in the Connection Broker

All past and present invitations appear on the **> Users > Collaboration** page, shown in the following figure.



The screenshot shows the Leostream Web Client interface. The top navigation bar includes links for Status, Resources, Clients, Plans, Users, System, and Search. The 'Users' tab is selected, and the 'Collaboration' sub-tab is active. Below the navigation bar, there's a table of shadowed sessions. The table has columns for Actions, Type, Status, ID, Session owner, Shadow user, Desktop, Sent, Viewed, and Declined. The 'Status' column shows 'Cancelled', 'Viewed', 'Declined', and 'Sent'. The 'Sent' column shows timestamps. The 'Viewed' column shows timestamps. The 'Declined' column shows timestamps. The 'Actions' column contains links like 'View' and 'View | Cancel'.

Actions	Type	Status	ID	Session owner	Shadow user	Desktop	Sent	Viewed	Declined
<a href="#">View</a>	Invitation	Cancelled	17	leo	kgondoly	kdg-ubuntu-NXserver	01/27/2012 - 15:36:38		
<a href="#">View</a>   <a href="#">Cancel</a>	Invitation	Viewed	18	leo	kgondoly	kdg-ubuntu-NXserver	02/07/2012 - 10:26:46	02/07/2012 - 11:11:04	
<a href="#">View</a>	Invitation	Declined	19	leo	kgondoly	kdg-ubuntu-NXserver	02/07/2012 - 11:16:18	02/07/2012 - 11:16:30	02/07/2012 - 11:16:30
<a href="#">View</a>   <a href="#">Cancel</a>	Invitation	Sent	20	leo	kgondoly	kdg-ubuntu-NXserver	02/07/2012 - 11:17:13		

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.

## Using NoMachine NX Collaboration in the Leostream Web Client

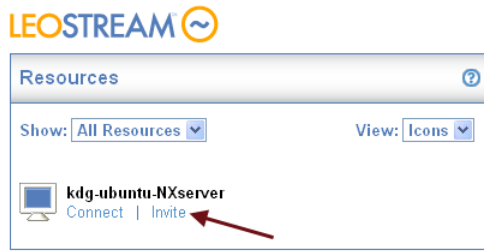
Users with the appropriate role and policy settings can share NoMachine NX sessions that are established from the Leostream Web client.

### ***Sending a Collaboration Invitation***


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

1. Click the **Connect** link to establish the NX session.
2. After the NoMachine NX sessions starts, click the **Refresh List** link in the Leostream Web client. All desktops with an established NX session that support collaboration now include an **Invite** link.

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



- In the **Send Invitation** form that opens, select the user to invite to the session from the **Shadow user** drop-down menu.

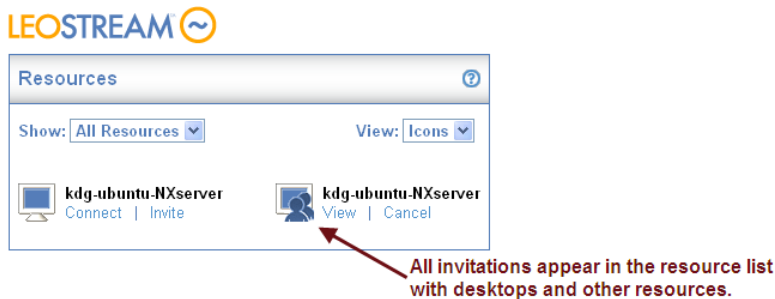
 The selected user *must* have a role that gives them permission to collaborate with other users. Otherwise, the user will not see the invitation.

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

 A screenshot of the 'Send Invitation' form. It has a title bar 'Send Invitation'. The form contains the following fields: 'Session owner' (kcondoly), 'Shadow user' (leo), 'Assigned desktop' (kdg-ubuntu-NXserver), 'Expire after' (10 minutes), and 'Notes' (Please, join my session so we can do a code review.). At the bottom are 'Send' and 'Cancel' buttons.

- Click **Send**.

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

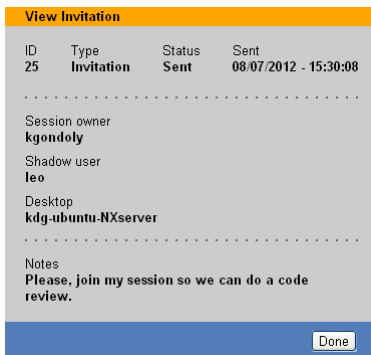


Select **Invitations** from the **Show** menu to limit the resource list to only sent and received invitations, as

shown in the following figure.

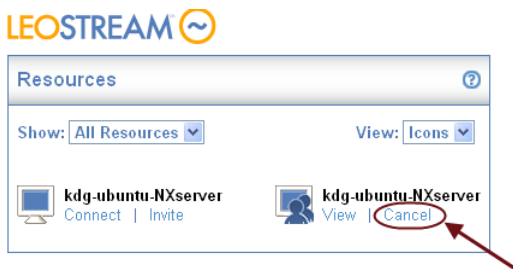


Click the **View** link associated with an invitation to see the details of that information, including any entered notes, for example:



### ***Cancelling an Invitation***

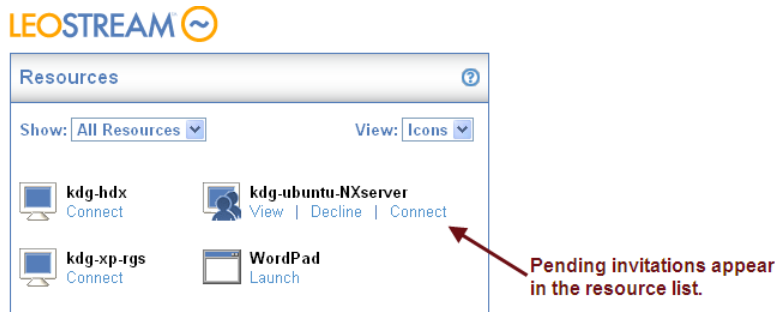
To cancel an invitation, click the **Cancel** link associated with that invitation, as shown in the following figure.



Invitations are automatically cancelled when the desktop is released back to its pool, for example, when you log out of the NX 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 and applications, 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.

## Teradici® PCoIP® Technology

The Leostream Connection Broker can initiate PCoIP connections to the following types of remote desktops.

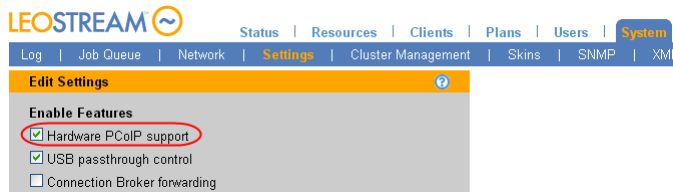
- Microsoft Windows and Linux workstations with installed PCoIP Remote Workstation cards. See the [Quick Start with Teradici PC-over-IP](#) for details on setting up the Connection Broker to manage PCoIP connections.
- Windows desktops with an installed PCoIP Agent, when using the Teradici Pervasive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

## Enabling PCoIP Remote Workstation Card Support

In order to manage workstations that have an installed PCoIP Remote Workstation Card, you must enable the global PCoIP feature, as follows.

1. Go to the > **System > Settings** page.
2. Select the **Hardware PCoIP support** option, shown in the following figure.



3. Click **Save**.
4. You must reboot the Connection Broker after enabling this feature, as follows:
  - a. Go to the **> System > Maintenance** page.
  - b. Select the **Reboot the Connection Broker** option.
  - c. Click **Next**.

You must sign back into the Connection Broker, after the reboot completes.

After you enable the PCoIP feature and reboot your Connection Broker, the Connection Broker adds the following items to the Web interface:

- 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 **> Resources > 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.

## PCoIP Connections to Workstations with a Remote Workstation Card

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.

## PCoIP Connections Using the Pervasive Computing Platform

For a complete description of how to use Leostream with the Teradici Pervasive Computing Platform, please download the following Leostream Quick Start Guide:

[http://www.leostream.com/hubfs/documentation/quick\\_start\\_Teradici\\_Pervasive\\_Computing\\_Platform.pdf](http://www.leostream.com/hubfs/documentation/quick_start_Teradici_Pervasive_Computing_Platform.pdf)

## PCoIP Connections to VMware Virtual Machines

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





When installing the Leostream Agent, ensure that you *do not* select the **End-user experience** 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 **> Plans > Protocol** 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 Chapter 12 of 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.

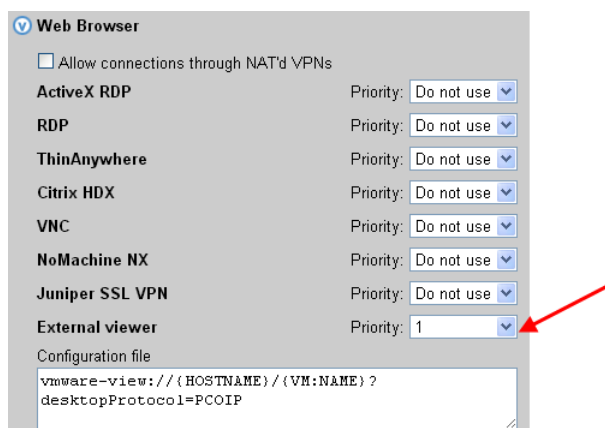
## Establishing Connections using the Leostream Web Client

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
```

The following figure displays the final protocol plan configuration.



The screenshot shows the 'Web Browser' configuration window. It has a checkbox 'Allow connections through NAT'd VPNs' which is unchecked. Below this is a list of protocols with their respective priorities: ActiveX RDP, RDP, ThinAnywhere, Citrix HDX, VNC, NoMachine NX, and Juniper SSL VPN, all set to 'Do not use'. The 'External viewer' is set to '1'. A red arrow points to the 'External viewer' priority dropdown. Below the list is a 'Configuration file' text area containing the URI: `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

Configure the connection to a device

TERADICI™ PCoIP® Zero Client

Session Connection Type: PCoIP Connection Manager ▼

Server URI: https://broker.yourcompany.net

4. Click **Apply**.



The zero client must be running Teradici firmware version 4.2, or higher.

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.

## Red Hat SPICE

Red Hat SPICE connections are available for virtual machines hosted by Red Hat Enterprise Virtualization for Desktops. For information on the SPICE protocol, consult the Red Hat Web site.


<http://www.redhat.com/virtualization/rhev/desktop/spice/>

To connect to desktops using SPICE, you must create a center for your Red Hat Enterprise Virtualization Manager. See "Understanding Connection Broker Centers" in the Connection Broker Administrator's Guide for information on creating the Red Hat center.

## Configuring the Client Device

In order to connect to a virtual machine using SPICE, the client device must include the following components.

- Leostream Connect version 2.8, or higher
- A SPICE client version 5.x

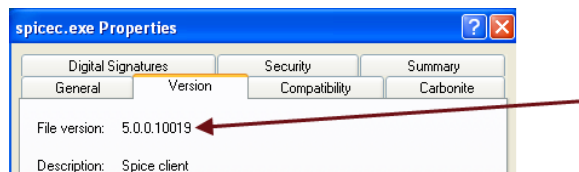
 Leostream Connect does not use the SPICE ActiveX component. You must ensure that `spicec.exe` exists on the client device and that users log in to the Connection Broker using Leostream Connect.


## Installing the SPICE Client

You must install version 5 of the SPICE client on each client device. After running the SPICE installer, you should find the `spicec.exe` file in a directory similar to the following.

`C:\Program Files\RedHat\RHEV\SpiceClient`

To ensure that you have the correct version of the SPICE client, open the **Properties** dialog for the `spicec.exe` file, go to the Version tab, and ensure that version 5 is installed, as shown in the following figure.



 Older versions of the SPICE client are not compatible with Leostream protocol plans. You must upgrade all `spicec.exe` files to version 5.

## Configuring a Connection Broker Protocol Plan for SPICE

After installing and copying the necessary files onto the client device, configure a Connection Broker protocol plan to establish SPICE connections.

1. Go to the **> Plans > Protocol** page.
2. Create a new protocol plan, or edit an existing plan.
3. In the **Leostream Connect and Thin Clients Writing to Leostream API** section of the protocol plan, scroll down to the **Red Hat SPICE** section, shown in the following figure.

4. By default, a new protocol plan is configured not to use the SPICE protocol. To create a plan that uses SPICE, change the **Priority** drop-down menu associated with SPICE to 1, as shown in the previous figure.
5. By default, RDP is set to a priority of 1. Before saving the form, you must ensure that no two protocols are assigned the same priority. Therefore, set the **Priority** drop-down menu associated with RDP to **Do not use**.
6. The **Command line parameters** edit field contains the following default value:

```
--host {HOST:IP} --port {HOST:PORT} --secure-port {HOST:SECURE_PORT} --secure-channels main,inputs --password {SPICE_TICKET}
```

The Connection Broker replaces the {HOST:IP}, {HOST:PORT} and {HOST:SECURE\_PORT} dynamic tags with the host name and ports used to connect to the Red Hat Enterprise Virtualization Manager server. The {SPICE\_TICKET} dynamic tag represents the secure ticket needed to establish communication between the SPICE client and host.

You do not need to edit the default command line parameters to establish a SPICE connection.

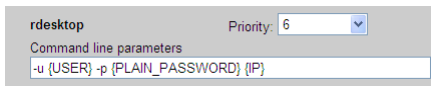
7. Save the protocol plan.

When you create Connection Broker policies for your users, ensure that you apply this protocol plan to pools of virtual machines that are hosted in a Red Hat Enterprise Virtualization 3.0 environment.

## 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, shown in the following figure.



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 and Ericom Blaze. 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 fullscreen 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 Mac OS 10, you must recompile rdesktop. Consult the FAQ in the Leostream Web site for more information.

## Sun Ray Options

### Sun Ray – uttsc

The **Sun Ray** section of the protocol plan, shown in the following figure, is near the bottom of the **Leostream Connect and Thin Clients Writing to the Leostream API** section.



By default, protocol plans are not configured to work in Sun Ray environments. To configure a protocol plan

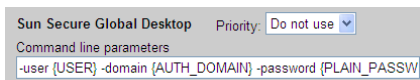
for Sun Ray deployments:

1. Change the **Priority** for Sun Ray to 1.
2. Change the **Priority** of all other protocols in the **Leostream Connect and Thin Clients Writing to the Leostream API** section to **Do not use**.
3. In the **Sun Ray** section, use the **Command line parameters** field to customize the rdesktop connection to the remote desktop.

See the Leostream [Thin Clients Guide](#) for a complete description of how to use Leostream in an Oracle Sun Ray™ environment.

## Sun Secure Global Desktop – ttatasc

The **Sun Secure Global Desktop** section of the protocol plan, shown in the following figure, is at the bottom of the **Leostream Connect and Thin Clients Writing to the Leostream API** section.



By default, protocol plans are not configured to work with Sun Secure Global Desktop (SGD). To configure a protocol plan for SGD deployments:

1. Change the **Priority** for Sun Secure Global Desktop to 1.
2. Change the **Priority** of all other protocols in the **Leostream Connect and Thin Clients Writing to the Leostream API** section to **Do not use**.
3. In the **Sun Secure Global Desktop** section, use the **Command line parameters** field to customize the Sun AIP connection. The final connection to the remote desktop is always done using Microsoft RDP.

See the “Sun Secure Global Desktop” in the [Connection Broker Administrator’s Guide](#) for complete instructions on setting up Leostream Connect to work in an SGD environment.



If you are using the latest SGD version, ensure that you remove the `-windowskey` on parameters from the **Command line parameters** in your protocol plans. The `windowskey` parameter is no longer supported when integrating with Leostream.

## VNC Remote Viewer

VNC is a viewer for Linux® and Windows NT4, 2000, XP, Vista, and Windows 7 operating systems. Leostream Connect supports four versions of VNC; RealVNC®, RealVNC Enterprise, TightVNC, and UltraVNC. UltraVNC allows the Windows username and password to be sent, enabling single sign-on.

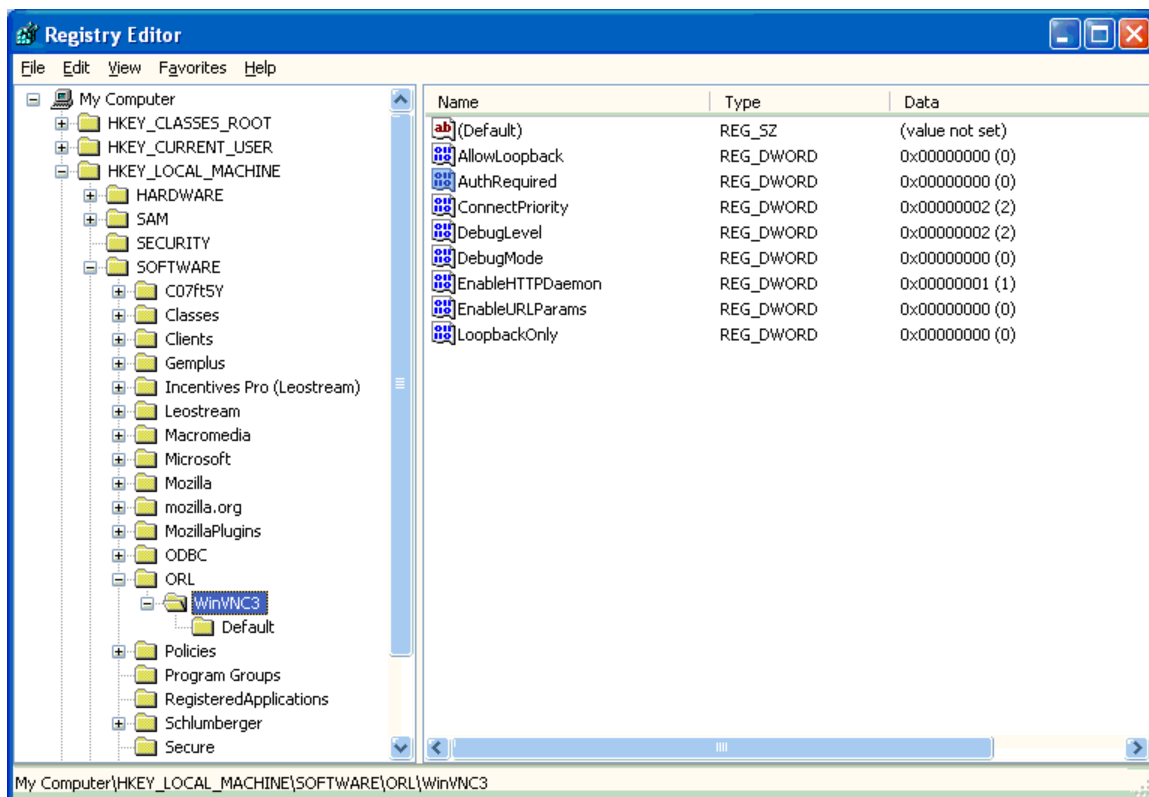


Single sign-on is support only for UltraVNC. To support single sign-on with VNC, you must select the single sign on task when installing the Leostream Agent on the remote desktops. Also, these users must have the **Enable single-sign-on to desktop console** option selected in the **When User is Assigned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

## Setting up VNC for Single Sign-On on Windows Operating Systems

The VNC server requires the client to supply a password, which is not the same as the user's Windows password. If you do not supply this password, before launching the VNC viewer, the VNC server opens a dialog for the end user to type the VNC password. If you do not want to provide your end users with the VNC password, disable the `AuthRequired` registry key on the VNC server, as follows.

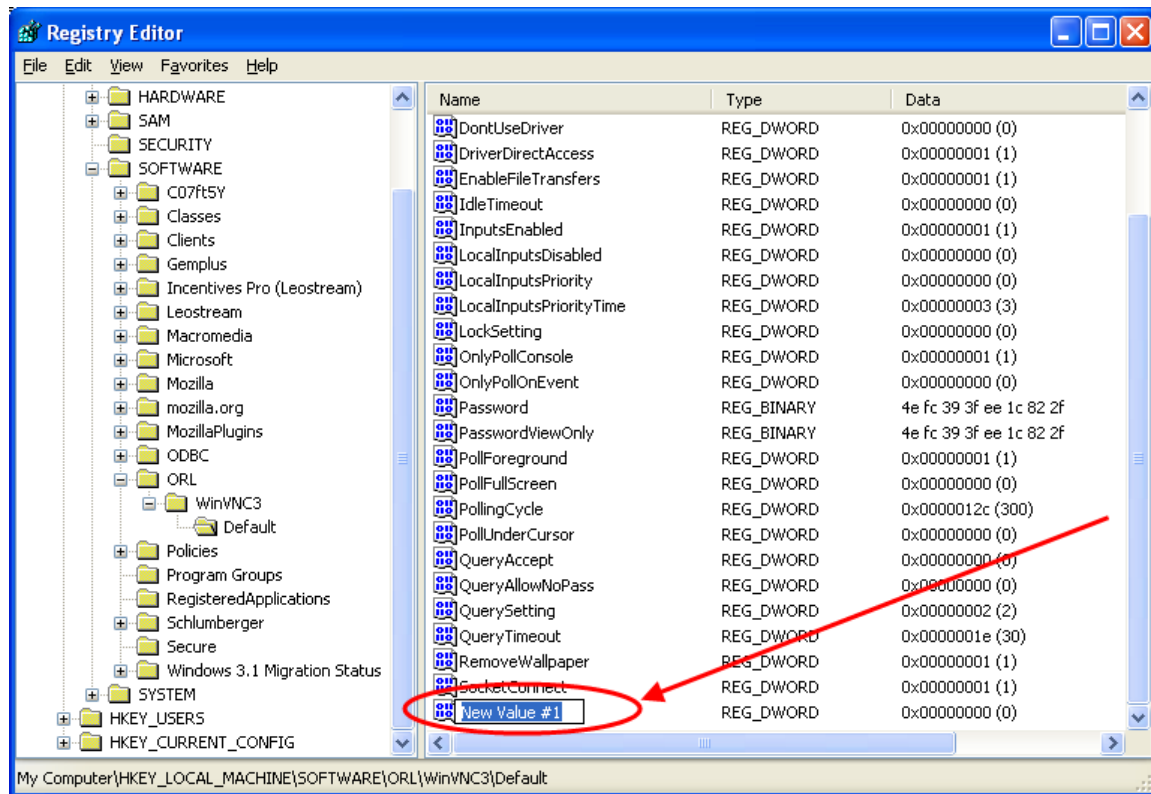
1. Run the `regedit` command to open the **Registry Editor** dialog.
2. Navigate to the `HKEY_LOCAL_MACHINE/SOFTWARE` folder.
3. Inside this folder, open the folder for your VNC installation, for example, `ORL/WinVNC3`, as shown in the following figure.



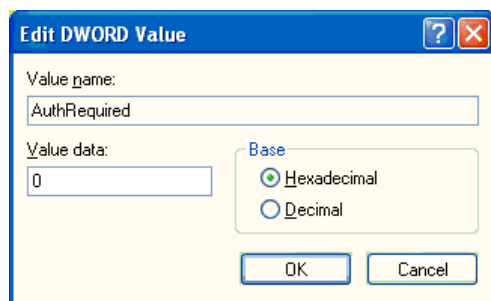
4. Select the `AuthRequired` key. If this key does not exist, create the key as follows:
  - a. Right-click in the list on the right side of the **Registry Editor** and select **New**.



- b. In the menu that opens, select **DWORD Value**. A new key appears in the list, as shown in the following figure.

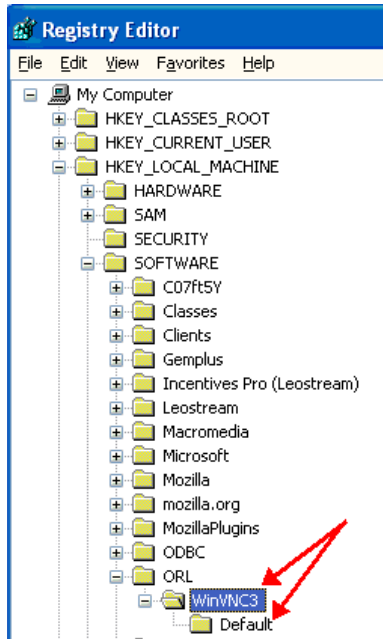


- c. Type the name `AuthRequired` into the new key.
  - d. By default, the new key takes the value 0. Keep this default.
5. To ensure that the `AuthRequired` key has the value of 0, right-click on the key and select **Modify**. The **Edit DWORD Value** dialog, shown in the following figure, opens.



6. Enter **0** for the **Value data**.
7. Click **OK**.

- Repeat step 4 through 7 for the **Default** folder inside the VNC folder, shown in the following figure.

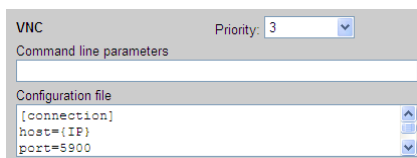


Restart the VNC service after you have set all the keys. The VNC server now accepts a null password. To set the password to null, go to the **Administrator Properties** for the VNC server and empty out the password field.


## Setting up the Connection Broker to Use VNC

To configure a protocol plan to use VNC

- Scroll down to the **VNC** section of the protocol plan, shown in the following figure.



- Select 1 from the **Priority** drop-down menu.
- Use the **Command line parameters** and **Configuration file** fields to customize the remote viewer session.

 The configuration file, by default, specifies port 5900 for VNC connections. This is the default VNC port when connection to a Windows desktop. If using VNC to connect to a Linux desktop, change the `port` parameter to 5901.

## 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<sup>®</sup> 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). Quality level 0 denotes bad image quality but very impressive compression ratios, while level 9 offers very good image quality at lower compression ratios. Note that the "tight" encoder uses JPEG to encode only those screen areas that look 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.

## RealVNC Enterprise Edition, UltraVNC, and TightVNC Configuration file

These versions of VNC support configuration files with a `.vnc` extension. The basic RealVNC does not provide support for a configuration file. For example:

```
[Connection]
UserName=David

Encryption=Server

SingleSignOn=1

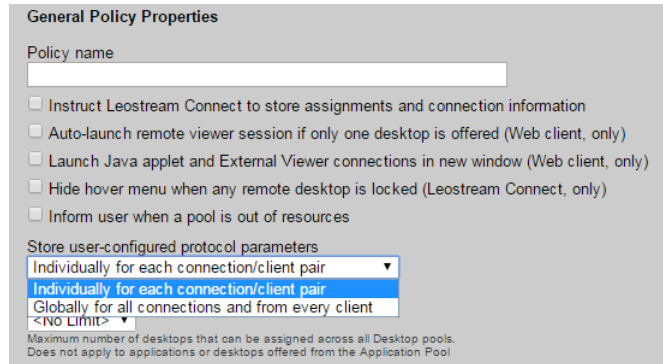
SelectDesktop=

[Options]
UseLocalCursor=1
UseDesktopResize=1
FullScreen=1
FullScreenChangeResolution=0
UseAllMonitors=0
RelativePtr=0
FullColour=1
LowColourLevel=1
PreferredEncoding SendKeyEvents=1
SendCutText=1
AcceptCutText =ZRLE
AutoSelect=1
Shared=0
SendPtrEvents=1
=1
ShareFiles=1
DisableWinKeys=1
Emulate3=0
```

## 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. User-configured 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.

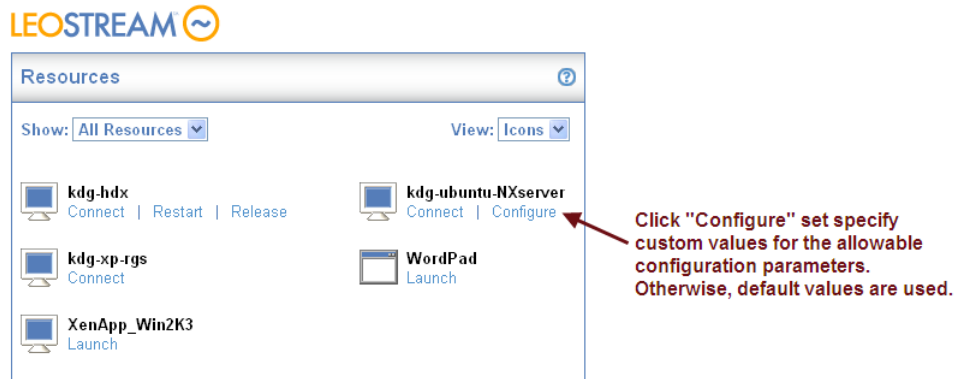
## End-User Interface for Configuring Parameters

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

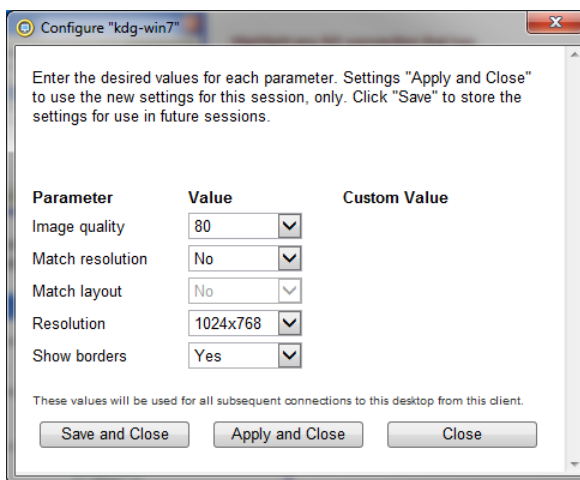
### Leostream Web Client

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.



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

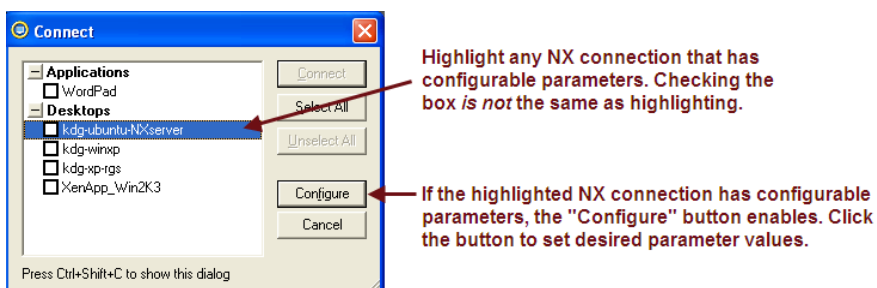


3. Clicks **Save and Close** to store new values for all subsequent applicable connections. Click **Apply and Close** to use the new values only for the current connection.

## 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, as shown in the following figure.



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

Configure "kdg-win7"

Enter the desired values for each parameter. Settings "Apply and Close" to use the new settings for this session, only. Click "Save" to store the settings for use in future sessions.

Parameter	Value	Custom Value
Image quality	80	
Match resolution	No	
Match layout	No	
Resolution	1024x768	
Show borders	Yes	

These values will be used for all subsequent connections to this desktop from this client.

Save and Close    Apply and Close    Close

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. For example, if the user is offered an NX and RGS connection from their policy, their **Configure** form may take the following form.



Configure connections

Enter the desired values for each parameter. Settings "Apply and Close" to use the new settings for this session, only. Click "Save" to store the settings for use in future sessions.

NoMachine NX Parameter	Value	Custom Value
Disable backingstore	False	
Resolution	1024x768	
Window manager	GNOME	

RGS Parameter	Value	Custom Value
Image quality	65	
Match resolution	No	
Match layout	No	
Resolution	1024x768	
Show borders	No	

These values will be used for all subsequent connections from this client.

Save and Close    Apply and Close    Close

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