



Connection Broker

Where Virtual Desktops Meet Real Business

Choosing and Using Display Protocols

April 17, 2012

Contacting Leostream

Leostream Corporation
411 Waverley Oaks Rd.
Suite 316
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.

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

Copyright

© Copyright 2002-2012 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.

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 products are patent pending.

Contents

CONTENTS	3
OVERVIEW	5
CHOOSING A DISPLAY PROTOCOL	5
THE PROTOCOL TRIANGLE.....	5
A QUESTION OF OPERATING SYSTEM	7
CONFIGURING DISPLAY PROTOCOLS IN LEOSTREAM	7
USING PROTOCOL PLANS	7
<i>How Protocol Plans Work</i>	8
<i>Building Protocol Plans</i>	9
SPECIFYING CONFIGURATION FILES AND COMMAND LINE ARGUMENTS.....	11
<i>Using Dynamic Tags in Configuration Files</i>	11
<i>Dynamic Remapping of Desktop IP Address</i>	14
<i>Client Dependent Variables</i>	15
CITRIX HDX	15
CONFIGURING THE CITRIX XENDESKTOP ENVIRONMENT	16
<i>Configuring the Citrix Desktop Delivery Controller</i>	16
<i>Creating a Citrix XenApp Services Site</i>	17
CONFIGURING CLIENT DEVICES	17
MAKING HDX CONNECTIONS TO RESOURCES ASSIGNED BY LEOSTREAM.....	17
<i>Step 1: Create a Citrix XenDesktop Center</i>	18
<i>Step 2: Define an HDX Protocol Plan</i>	19
<i>Step 3: Use the HDX Protocol Plan in Policies</i>	19
MAKING HDX CONNECTIONS TO RESOURCES ASSIGNED BY CITRIX.....	20
<i>Step 1: Enable the Citrix XenApp Services Site Feature</i>	20
<i>Step 2: Configure the Policy</i>	21
CITRIX ICA	21
USING THE CITRIX ONLINE PLUGIN.....	22
<i>Launching Desktop Connections in Fullscreen</i>	23
USING THE CITRIX CLIENT FOR JAVA.....	23
<i>Uploading New Client Versions</i>	24
<i>Launching the Client in a new Window</i>	24
ERICOM® BLAZE	24
FAMATECH RADMIN® 2.2 AND 3.X REMOTE VIEWER	26
HP® REMOTE GRAPHICS SOFTWARE (RGS)	26
<i>HP RGS Receiver Configuration Files</i>	26
<i>Multi-Monitor Support with HP RGS</i>	27
<i>Remembering Window Position for HP RGS Connections</i>	27
<i>Single Sign-On with HP RGS</i>	28
<i>USB Passthrough with HP RGS</i>	29
MICROSOFT® REMOTEFX	30
MICROSOFT® RDP REMOTE VIEWER	30
<i>Options for Encoding Desktop Login Credentials into RDP Configuration Files</i>	31

<i>Microsoft RDP Viewer Command Line Parameters</i>	31
<i>Microsoft RDP Viewer Configuration File Variables</i>	31
NOMACHINE NX CLIENT	39
LAUNCHING NX CONNECTIONS FROM THE WEB CLIENT	39
NX CONFIGURATION FILE	40
<i>Login Group</i>	40
<i>General Group</i>	41
<i>Advanced Group</i>	44
<i>Images Group</i>	44
SETTING USER-CONFIGURABLE PARAMETERS	45
<i>Enabling End-User Configurable Parameters</i>	45
<i>End-User Interface for Configuring Parameters</i>	46
SESSION SHADOWING AND COLLABORATION.....	48
<i>Configuring Collaboration in the Connection Broker</i>	48
<i>Managing Shadowed Sessions in the Connection Broker</i>	50
RED HAT SPICE	51
CONFIGURING THE CLIENT DEVICE.....	51
<i>Installing the SPICE Client</i>	51
CONFIGURING A CONNECTION BROKER PROTOCOL PLAN FOR SPICE	51
RDESKTOP RDP REMOTE VIEWER	52
SUN RAY OPTIONS	53
SUN RAY – UTTSC	53
SUN SECURE GLOBAL DESKTOP – TTATSC.....	53
VNC REMOTE VIEWER	54
<i>Setting up VNC for Single Sign-On on Windows Operating Systems</i>	54
<i>Setting up the Connection Broker to Use VNC</i>	57
<i>VNC Command Line Parameters</i>	57
<i>RealVNC Enterprise Edition, UltraVNC, and TightVNC Configuration file</i>	59

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
- Oracle Appliance Link Protocol (ALP)
- Oracle Adaptive Internet Protocol (AIP)
- 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.

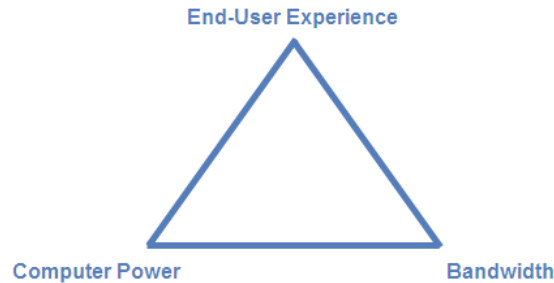
The performance and requirements of the remote session are, therefore, completely determined by the display protocol you select. This chapter tries to provide some food-for-thought when investigating and choosing from the virtual soup of display protocols.

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 chart provides an overview of how some popular display protocols rank for each angle in the protocol triangle.

Protocol	Compute Power	Bandwidth	User Experience	Comments
Microsoft RDP	Low	Medium	Medium	Not ideal for networks with high latency
Citrix ICA	Low	Low	Medium	Good in high latency environments
HP RGS	High	Low	High	Good in bladed environments running on a LAN with adequate bandwidth. Not suitable for WAN.
Teradici PCoIP (hardware)	Very High	High	Very High	The compute power is provided by the dedicated Teradici chips
VMware/Teradici PCoIP (software)	Very High	High	Very High	The compute power comes either from more powerful client devices or by placing fewer VMs on each server
NoMachine NX	Medium	Low-Medium	Medium - High	Particularly well suited to challenging network environments; require an NX client on the client device

You can also use the following questions to refine 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 ICA	Linux	No	Yes
	Windows	No	Yes
Ericom Blaze	Linux	No	Yes
	Windows	No	Yes
Famatech Radmin	Linux	No	Yes*
	Windows	No	Yes
HP RGS	Linux	Yes	Yes
	Windows	Yes	Yes
Microsoft RDP	Linux	No	No
	Windows	No	Yes
NoMachine NX	Linux	Yes	No**
	Windows	Yes	No**
rdesktop	Linux	No	Yes
	Windows	No	No
VNC	Linux	Yes	Yes
	Windows	Yes	Yes

* 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.

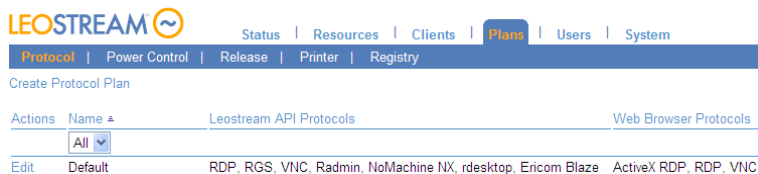
For thin clients running a vendor-supplied operating system, please consult your thin client vendor.

Configuring Display Protocols in Leostream

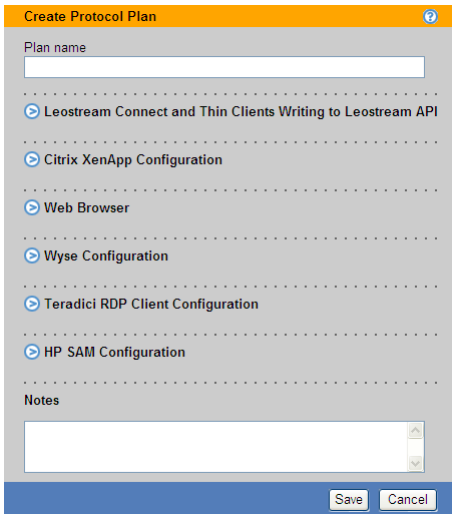
Using Protocol Plans

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 remote viewer used when the user logs in from Leostream Connect, thin clients, PCoIP enabled clients, the Leostream Web client, and HP SAM clients. You configure the remote viewer for each of these clients separately, using the appropriate section in the protocol plan, shown collapsed in the following figure.

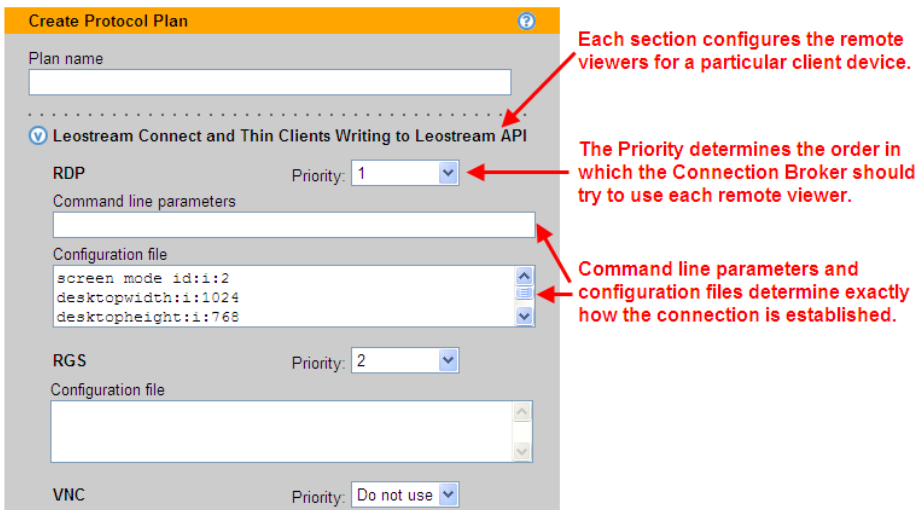


How Protocol Plans Work

Protocol plans give you the flexibility to configure which of these protocols to use for each pool used in each 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 selection in the **Priority** drop-down menu indicates the order in which the Connection Broker tries to establish a connection using that remote viewing 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. When the Connection Broker runs out of remote viewers to try, i.e., the **Priority** drop-down menu for all other remote viewers in the protocol plan 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 remote viewer can be used, the Connection Broker performs a port check. 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 remote viewers 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

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".

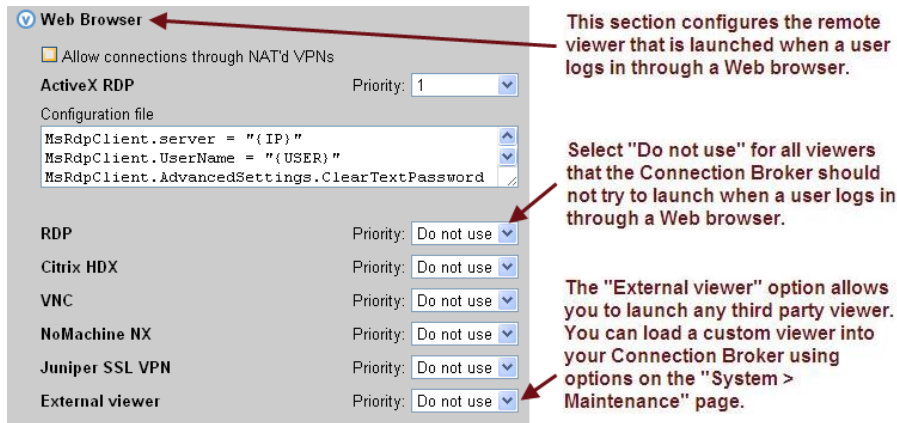
5. In the **Citrix XenApp (ICA) Configuration** section, shown in the following figure, configure the command line parameters and ICA-file to use when launching a desktop or application published in a Citrix XenApp farm. This section applies to users logging in from any of the following client devices

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

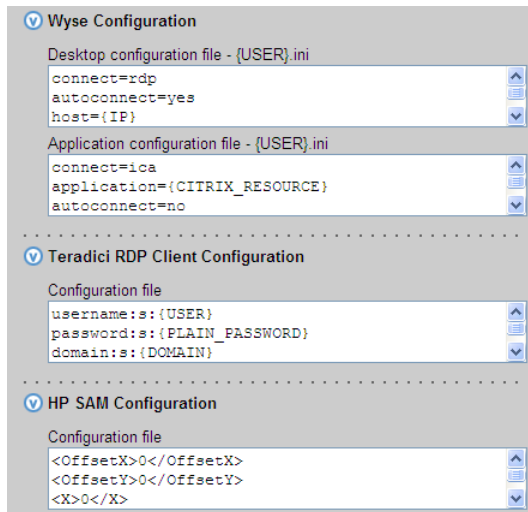
Select this option to launch Citrix XenApp resources without requiring that a Citrix client be installed on the client device.

6. In the **iPhone and iPad Devices Connecting with iTap** section, shown in the following figure, configure the display protocols to use when a user logs in to Leostream from their mobile device, using the iTap RDP client.

7. 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.



8. 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 enabled clients that also support RDP
 - HP SAM clients



9. Use the **Notes** field to store any additional information with your protocol plan.
10. 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 specific 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.

Guide to Choosing and Using Display Protocols

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.
{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.
{USER}, {USER:USER}, {USER:LOGIN_NAME}, or {LOGIN)NAME}	The user's login name. This value corresponds to the value shown in the Login name column on the > Users > Users page.
{NAME} or {USER:NAME}	The user's display name. This value corresponds to the value shown in the Name column on the > Users > Users page.
{AD_DN} or {USER:AD_DN}	The user's Active Directory Distinguished Name. This value corresponds to the value shown in the AD Distinguished Name column on the > Users > Users page.
{EMAIL} or {USER:EMAIL}	The user's email address. This value corresponds to the value shown in the Email column on the > Users > Users 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.
{DOMAIN}	The name entered into the Domain field for the authentication server that authenticated a user. If the Domain field is empty, the Connection Broker replaces this dynamic tag with the authentication server name.
{AUTH_DOMAIN}	The name entered in the Authentication server name 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 ticketed 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 Name column on the >

Dynamic Tags	Purpose
	Clients > Clients 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 IP Address column on the > Clients > Clients 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 MAC Address column on the > Clients > Clients page.
{CLIENT:TYPE}	The type of client used to log into the Connection Broker. This value corresponds to the value shown in the Type column on the > Clients > Clients page.
{CLIENT:MANUFACTURER}	The manufacturer of client used to log into the Connection Broker. This value corresponds to the value shown in the Manufacturer column on the > Clients > Clients page.
{CLIENT:UUID}	The UUID of the client used to log into the Connection Broker. This value corresponds to the value shown for the Client UUID on the Edit Client page.
{POOL:NAME}	The name of the pool that contains the desktop that the user is connecting to
{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 {ENV: HTTP_COOKIE} might return <code>uid=25157202</code> .
{MATCHED_IP:partial_IP_address}	<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 Tags	Purpose
{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 Remapping of Desktop IP Address

You can enable remote viewer traffic to traverse one or more NATed firewalls by dynamically changing the IP address provided to the remote viewer 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 take 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 IP address of the client 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 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.

Leostream can establish HDX connections to desktops assigned by the Leostream Connection Broker or by a Citrix XenDesktop farm's Desktop Delivery Controller (DDC). In cases where Leostream assigns the desktop, Leostream establishes the HDX connection by creating a Desktop Group in the Citrix Desktop Delivery Controller, assigning the user to that Desktop Group, and establishing the connection using the XenApp Services Site that publishes the resources in that DDC.

 Leostream currently does not support XenDesktop 5.x. Contact sales@leostream.com for information on when XenDesktop 5.x support will become available.

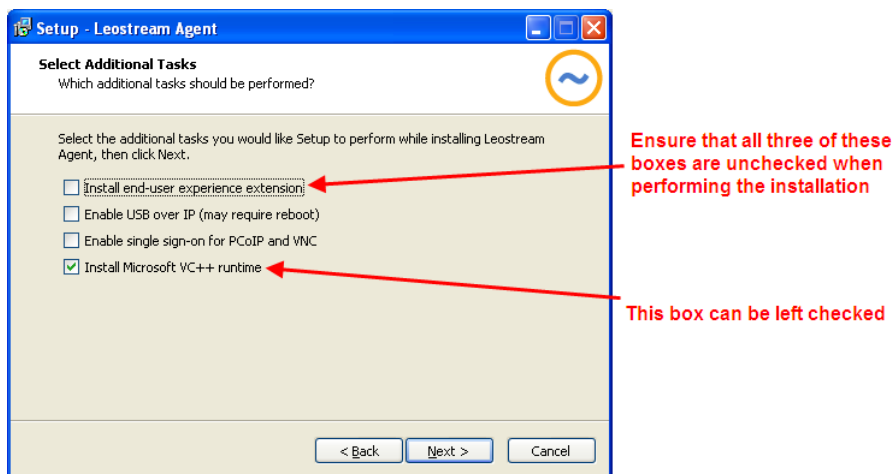
Configuring the Citrix XenDesktop Environment

Configuring the Citrix Desktop Delivery Controller

 This section does not apply if you plan to have the Connection Broker pull resources that are already assigned to a user in a DDC. Instead, use the **Desktop Assignment from Citrix XenApp Services Site** section of user's policy to indicate which XenApp Services site offers the user's XenDesktop resources; see the "Offering Resources from a Citrix XenApp Services Site" section of the [Connection Broker Administrator's Guide](#) for complete instructions.

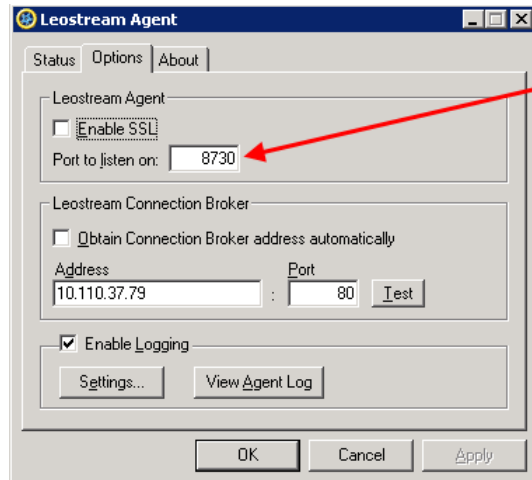
If you plan to have Leostream create Desktop Groups in the Citrix DDC and push desktop assignments into Citrix, you must create a XenDesktop center in your Connection Broker. In order to create a XenDesktop center, you must

- Install a Leostream Agent on the Desktop Deliver Controller. When installing the Leostream Agent, on the third page of the installation wizard, ensure that no additional features are installed, as shown in the following feature. 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 the DDC. Leostream recommends configuring the Leostream Agent to use port 8730, as follows.

1. On the DDC machine, 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.



- Install the Citrix Desktop Delivery Controller SDK on your XenDesktop DDC. Consult the following Citrix Developer Network post for download and installation instructions:

<http://community.citrix.com/display/xd/Download+SDKS>

- After you install the SDK, 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

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:



You must create a XenApp Services Site that includes your XenDesktop server farm.

You will use this Site URL to integrate the Services Site into the Leostream Connection Broker.

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

Configuring Client Devices

You must install the following two applications on each client device that connects to a remote desktop using HDX.

1. On each client device that establishes an HDX connection, install Leostream Connect. HDX connections are currently supported only when logging in through Leostream Connect.
2. On each client device, install the Citrix online plug-in. If users are currently establishing HDX connections from the client, the Citrix online plug-in is typically already installed.

Making HDX Connections to Resources Assigned by Leostream

In this scenario, Leostream *pushes* desktop assignments into the Citrix Desktop Delivery Controller.

Step 1: Create a Citrix XenDesktop Center

After the Leostream Agent is installed on the DDC, 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 from the **Type** drop-down menu. The form updates, as follows:

Add Center

Type
Citrix XenDesktop
If you change the type please wait for the form to repaint

Name

Desktop Delivery Controller address

Agent RPC port
8080

Folder for Leostream assignments
Leostream Desktops
A Folder with this name will be created in the DDC, and all desktops assigned by Leostream will be in Desktop Groups within this folder

Username
Username may be specified as "DOMAIN\USER"

Password

Refresh interval
10 minutes

Notes

Save Cancel

Enter a display name for this center.


Enter the IP address of the primary Desktop Delivery Controller (DDC) in your XenDesktop farm

You must install the Leostream Agent on the Citrix DDC. Enter the port number that the Leostream Agent listens on. The default port is 8080. To avoid conflicts, change the Leostream Agent port, for example, to 8730.

Specify a folder to create in the DDC, which then holds all the Leostream assignments. Do not manually create this folder. The Connection Broker automatically creates the folder when you save the center.

Enter the username and password for a user with administrator rights to the desktop where the DDC is installed. The username must include the user's domain, for example leostream\admin.

4. Enter a name for the center in the **Name** edit field.
5. In the **Desktop Delivery Controller address** edit field, enter the IP address of the primary Desktop Delivery Controller (DDC) in your XenDesktop farm.
6. In the **Agent RPC port** edit field, enter the port that the Leostream Agent installed on the DDC listens on. Ensure that this port is different from any of the ports used by the DDC.
7. In the **Folder for Leostream assignments** edit field, enter the name of the folder you want to create in the DDC. The Connection Broker places all new Desktop Groups into this folder.

 Do not manually create this folder. The Connection Broker automatically creates the folder when you save the **Create Center** form.

8. In the **Username** edit field, enter the username for a user that has administrator rights to the desktop where the DDC 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 successfully save the center (the center is listed as *Online* on the **> Resources > Centers** page), the Connection Broker automatically creates a folder in the Citrix Desktop Delivery Controller. This new folder has the name you specified in the **Folder for Leostream assignments** edit field.

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.

The screenshot shows a configuration form for 'Leostream Connect and Thin Clients Writing to Leostream API'. It includes two priority dropdown menus: 'RDP and RemoteFX' (set to 'Do not use') and 'Citrix HDX' (set to '1'). Below these are instructions to configure two fields: 'Create assignment in selected XenDesktop center' (a dropdown menu) and 'Site URL for XenApp Services Site' (a text field). A note at the bottom provides an example URL: `http://192.168.0.1/Citrix/PNAgent`.

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

Ensure that this DDC does not contain Desktop Groups that assign 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 DDC select in the previous step.

This URL takes the following form:

`http://192.168.226.133/Citrix/PNAgent`

4. Ensure that no other protocol in the section select **1** for their priority.
5. Click **Save** to save the protocol plan.

Step 3: 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 appropriate pools, 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

After a user with this policy logs in to the Leostream Connection Broker and requests a connection to a desktop from this pool, the Connection Broker performs the following steps.

1. The Connection Broker automatically creates a new Desktop Group in the Citrix Desktop Delivery Controller. The name of this Desktop Group begins with the name of the desktop the user is connecting to, followed by a unique ID.
2. The Connection Broker then adds the user's desktop to this Desktop Group, and assigns the group to the user. The Connection Broker creates a new desktop group for every requested HDX connection.
3. The new Desktop Group exists for the length of time the desktop is assigned to the user. As soon as the Connection Broker releases the desktop back into its pool, the Connection Broker deletes the Desktop Group from the DDC.

Citrix XenDesktop restricts a desktop to be a member of a single Desktop Group. Therefore, ensure that your Desktop Delivery Controller does not define any Desktop Groups that already contain the desktops you want to assign in Leostream.

Making 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 online plug 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.



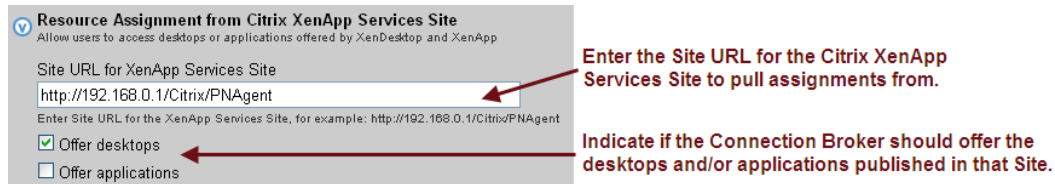
Step 2: Configure the Policy

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:

`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 fat desktop, laptop, or thin client running Leostream Connect, you must install the Citrix Online Plugin on the client device. Users logging in from the Leostream Web client can use 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=HTTPonTCP
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"
    >
  </body>
</html>
    
```

```

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

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

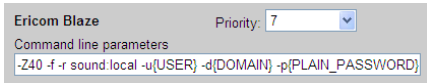
Ericom **Blaze** provides software acceleration for Microsoft RDP connections. Blaze is available when connecting from the Windows and Java versions of Leostream Connect.

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

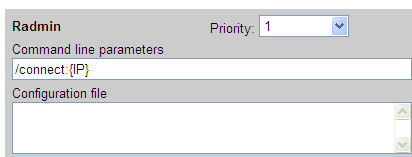
For all platforms

- 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.

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)

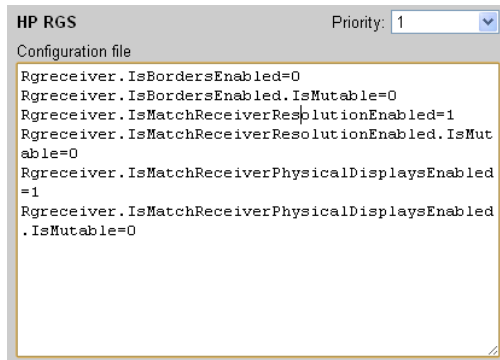
HP® Remote Graphics Software (RGS) is a high performance remote graphics system that renders the graphics on the desktop and sends the resulting screen image to the remote client. To learn more about HP RGS, refer to the [HP RGS user manual](#).

✓ Leostream Connect has been fully qualified with HP RGS version 5.2, 5.3, and 5.4.

To ensure that the Connection Broker establishes an HP RGS connection, switch the **Priority** for RGS to 1.

HP RGS Receiver Configuration Files

Use the **Configuration file** field associated with RGS to specify RGS Receiver properties, as shown in the following figure.



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

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 Chapter 8 “RGS properties” in 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
```

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.

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.

The image shows two screenshots from the Leostream Connect interface. The left screenshot is titled 'Create Protocol Plan' and shows the configuration for a protocol plan named 'Track RGS Windows'. It includes sections for RDP (Priority: 2) and RGS (Priority: 1). The RGS configuration file contains the following text:


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

 The right screenshot is titled 'Create Policy' and shows the configuration for a policy named 'RGS'. It includes sections for 'General Policy Properties', 'Desktop Assignment from Pools', 'When User Logs into Connection Broker', 'When User is Assigned to Desktop', and 'Plans'. The 'Plans' section shows the 'Track RGS Windows' protocol plan assigned to the policy.

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.

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, you must use the HP USB redirector. You can use the Leostream Connect sidebar to select which active remote session has access to all USB devices.

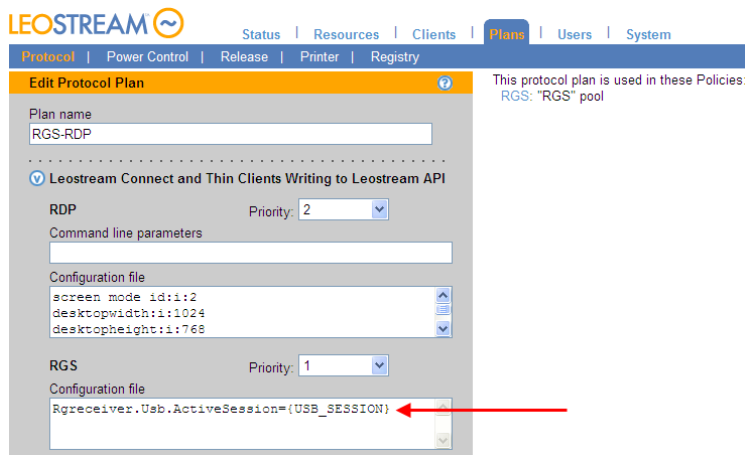
To turn on the sidebar for USB access:

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

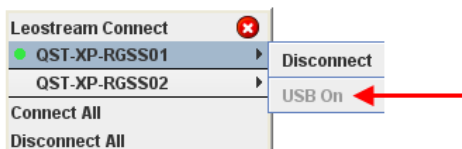
```
sidebar_enabled = true
```

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}
```



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.

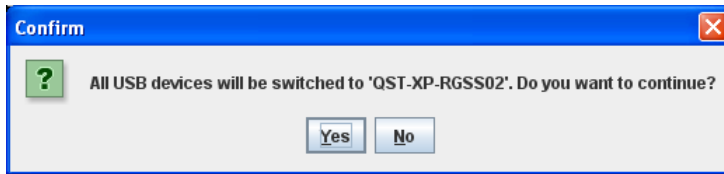


When you attach a USB device to your client device, the USB device appears in the remote desktop that indicates **USB On**. You can switch all USB devices to another desktop by selecting the **Turn USB On** menu associated with that desktop, as shown in the following figure.



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.

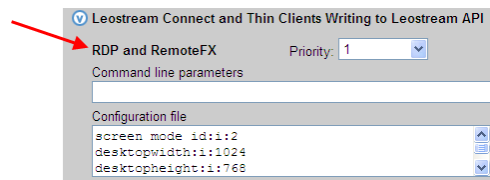
Microsoft® RemoteFX

Microsoft RemoteFX is a set of end-user experience enhancements for RDP. Leostream seamlessly supports RemoteFX using the same mechanisms used for RDP. To establish a RemoteFX connection, set the necessary parameters in the RDP-file in the user's protocol plan. The Connection Broker establishes a RemoteFX connection if the client and remote desktop support RemoteFX, otherwise RDP is used.

See the following section for information on the RDP and RemoteFX section of Connection Broker protocol plans.

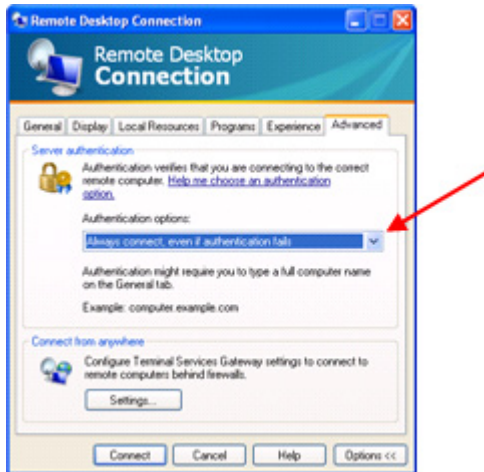
Microsoft® RDP Remote Viewer

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 RDP 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.



If you are running RDP version 6.0 or 7.0 on your Microsoft Windows® XP or Vista® local desktop, to ensure the remote viewer functions properly, do the following:

1. Open the RDP viewer **Options** dialog.
2. Go to the **Advanced** tab.
3. In RDP 6, select the **Always connect, even if authentication fails** option in the drop-down menu in the **Server authentication** section, shown in the following figure.



In RDP 7, select the **Connect and don't warn me** option from the drop-down menu.

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}
```

Associating desktop login credentials with policies instead of users allows the desktop login credentials to be coded into the RDP configuration file. This feature supports environments such as customer support, classrooms, and QA, where users pick a desktop from a pool of desktops that all have the same password.

Use the `{STANDARD_RDP_PASSWORD:password}` dynamic tag to pass an unencrypted password down to the Leostream Connect client in order to enable single sign-on. Replace `password` with the password to log into the desktop.

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-765333905911033.mspx>



Microsoft RDP 7.0 clients are now available for Windows XP and Vista. You can download RDP 7.0 from the Microsoft support site (see the Microsoft knowledgebase article [969084](#)).

/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 useful RDP configuration file parameters. 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://msdn.microsoft.com/en-us/library/ms861803.aspx>

<http://support.microsoft.com/?kbid=885187>

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

smart sizing:i:

Determines if the remote session is opened with or without scrollbars. When smart sizing is on, the entire remote session is always visible in the client window, with no scrollbars. If `screen mode` is set to 2, smart sizing is not necessary.

Value	Setting
0	Smart sizing off
1	Smart sizing on

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.

authentication level:i:

Indicates if RDP should open a dialog to ask if you trust the computer you are trying to remotely log into.

Value	Setting
0	No authentication
1	Require authentication
2	Attempt authentication

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:l

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

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

autoreconnection enabled:i

Determines if a client computer automatically tries to reconnect after being disconnected. This setting corresponds to the selection in the **Reconnect if Connection is dropped** check box on the **Experience** tab of Remote Desktop Connection **Options** dialog, according to the following values.

Value	Setting
0	Client computer does not automatically try to reconnect
1	Client computer automatically tries to reconnect

prompt for credentials:i:0

Determines if the user is prompted for their credentials, according to the following values.

Value	Setting
0	Always prompt for credentials
1	Do not prompt for credentials (requires EnableCredSSPSupport)

EnableCredSSPSupport:i:0

Determines if the user is prompted for their credentials, according to the following values..

Value	Setting
0	Do not prompt for credentials
1	Prompt for credentials

NoMachine NX Client

The NX Client allows your end users to access Linux desktops, and applications hosted on a Linux desktop. See the following NoMachine Web page for information on supported applications.

<http://www.nomachine.com/supported-applications.php>

To use NoMachine NX:

1. Install the NX Client, NX Node, and NX Server packages on the Linux VMs. The packages must be installed in this order. Ensure that SSH is also installed on the VM
2. Install the NX Client on the client machine.

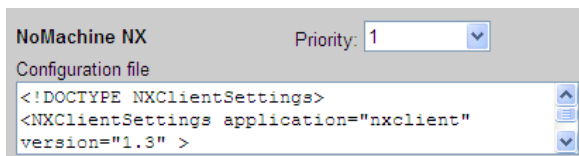
For installation instructions, go to:

<http://www.nomachine.com/installation.php>

Refer to the [NoMachine Documents](#) Web page for complete information on installing and configuring NX Servers and NX Clients.

Once all components are installed, configure the protocol plan to use NX. You can configure the protocol plan to use NX to connect to the desktop when the user logs in from Leostream Connect or the Leostream Web client.

1. Scroll down to the **NoMachine NX** section of the protocol plan, shown in the following figure, in the **Leostream Connect and Thin Clients Writing to Leostream API and Web Browser** sections.



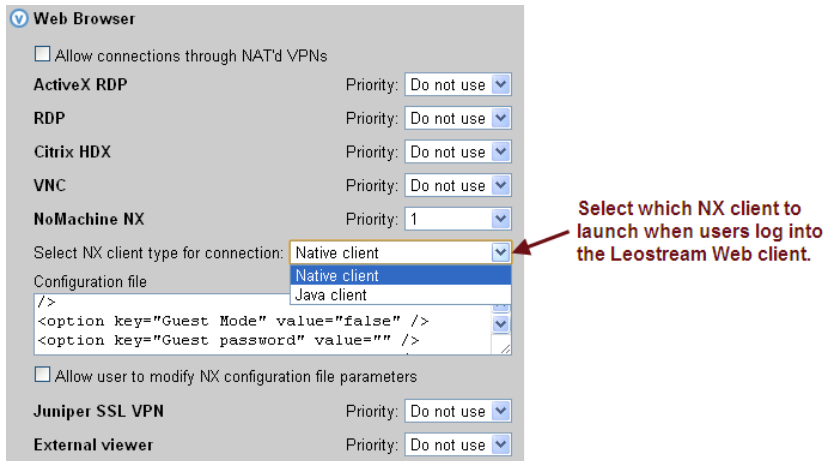
2. Change the **Priority** to 1.
3. If another protocol in the associated section of the protocol plan has a priority of 1, modify that protocol's priority to a lower number or to **Do not use**.
4. Use the **Configuration file** field associated with NX to customize the remote session. The configuration file specifies parameters in the `.nxs` file. These parameters map to entries in the NX Client GUI, as discussed in [NX Configuration File](#).
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.

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 Java applet. 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.

- 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.



- 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.
- Use the **Configuration file** field to define NX parameters for the connection. This field applies to the native NX client and NX Java applet.

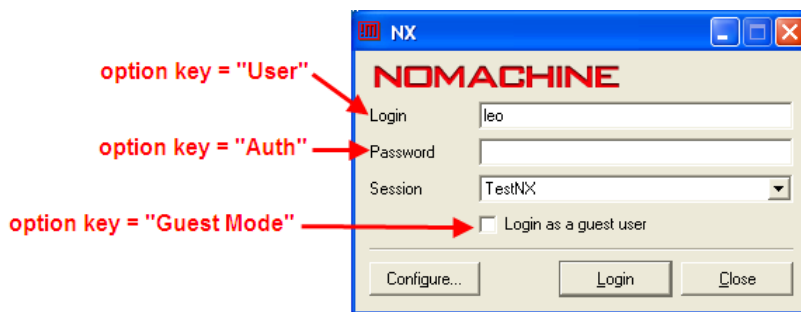
NX Configuration File

The configuration file for NX is an XML representation of the fields in the NX Client GUI. Each group in the XNL-file corresponds to the fields on a particular panel in the client's GUI.

The following figures and tables indicate how many of the XML-file maps to the fields in the NX Client GUI. For a complete description of the NX Client GUI, see the [NX Client Guide](#).

Login Group

The following figure indicates option keys in the configuration file that correspond to fields on the **Login** dialog. The **Session** drop-down menu does not have a corresponding entry in the configuration file. When using NX outside of Leostream, the session indicates the name of the NXS-file.

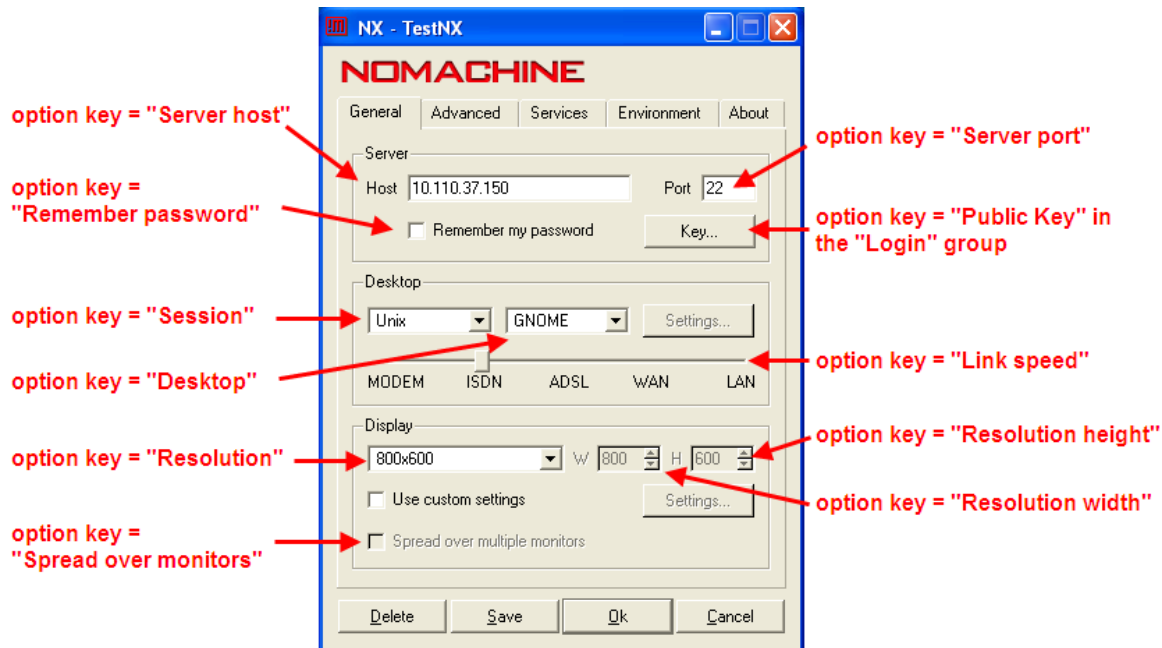


The following table describes the important option keys in the `Login` group when integrating with Leostream.

Option Key	Dialog	Purpose
User	Login	(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	Login	(Default = {SCRAMBLED_PASSWORD}) The user's password. By default, the configuration file is configured to pass a scrambled password. If your NX server is configured to expect a plain password, replace the {SCRAMBLED_PASSWORD} dynamic tag in the Configuration file field with the {PLAIN_PASSWORD} dynamic tag. When the {SCRAMBLED_PASSWORD} is used, the Connection Broker uses the NoMachine method for scrambling passwords. This scrambled password is then passed in the configuration file.
Guest Mode	Login	(Default = false) Indicates if the user should be logged in using the NX guest account on the remote desktop.
Guest password	Login	(Default = empty) If logging in as the guest user, enter in the password for the guest account.
Guest username	Login	(Default = empty) If logging in as the guest user, enter in the user name for the guest account.
Public Key	General	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. <pre> ... <option key="Login Method" value="nx" /> <option key="Public Key" value="" -----BEGIN DSA PRIVATE KEY----- < Insert DSA Key here> -----END DSA PRIVATE KEY----- /> ... </pre>

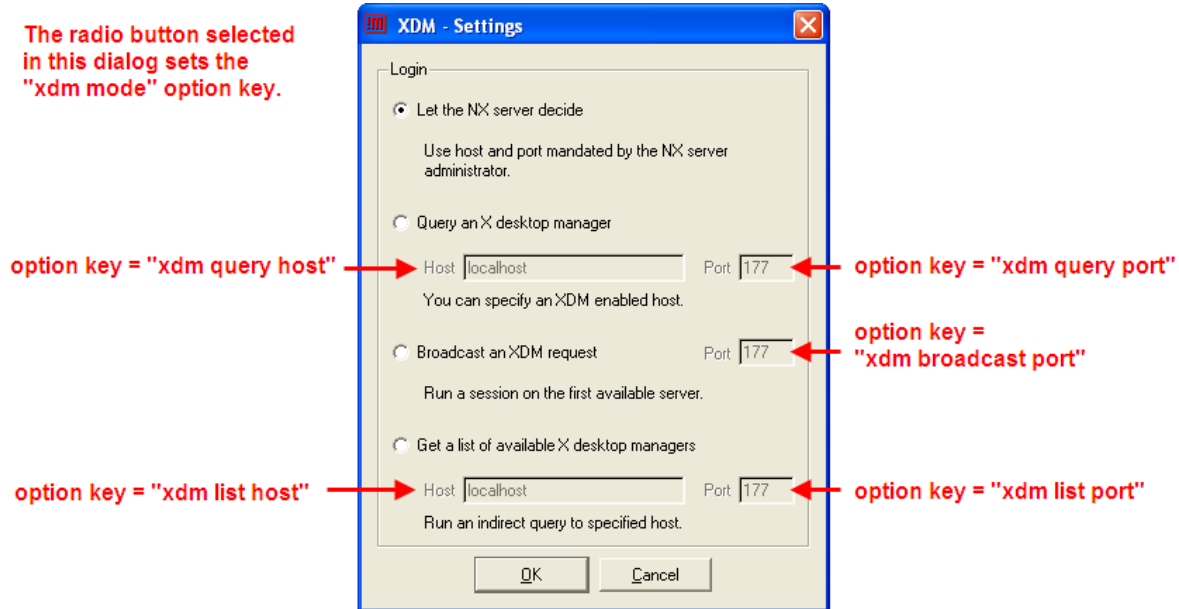
General Group

The following figures indicate option keys in the configuration file that correspond to fields on the **General** tab in the **Configuration** dialog.

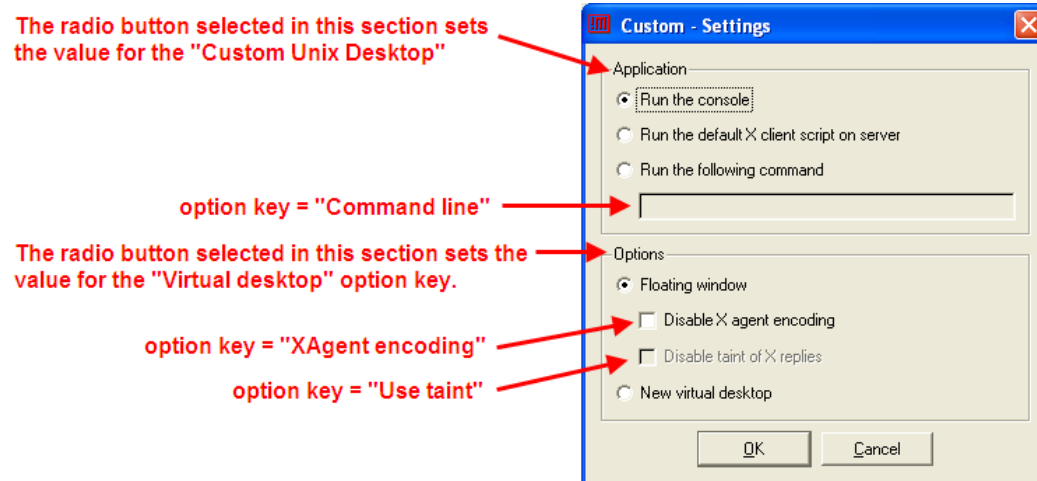


Additional Desktop Settings

Selecting **XDM** for the desktop type enables the **Settings** button. You can then specify the option keys shown in the following figure.

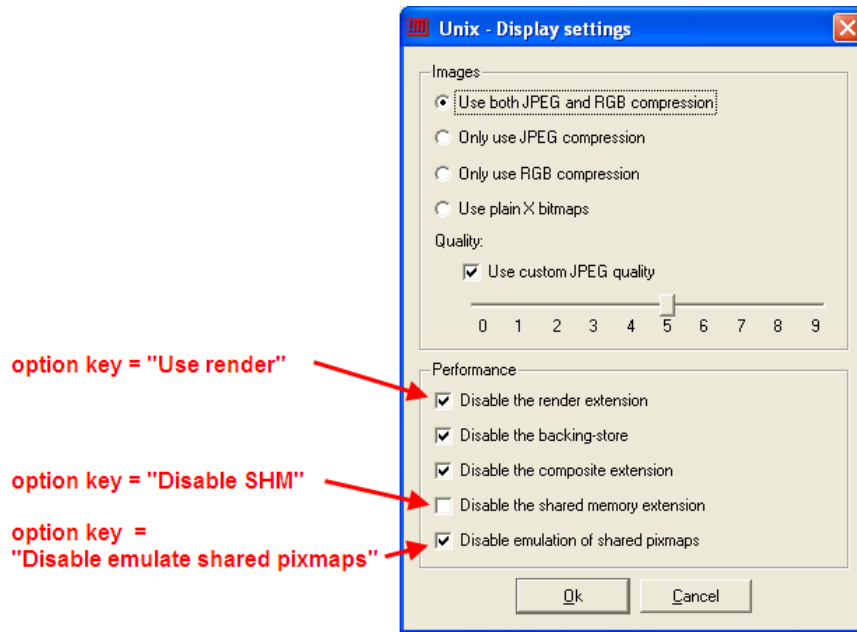


Selecting **Custom** for the desktop type enables the **Settings** button. You can then specify the option keys shown in the following figure.



Additional Display Settings

Selecting **Use custom settings** in the **Display** section enables the **Settings** button. Clicking the **Settings** button opens one of three dialogs, depending on your selection in the **Session** drop-down menu. Most of the fields on these dialogs correspond to option keys in the **Images** group (see **Images Group**). The following figure indicates the options that correspond to the **General** group.

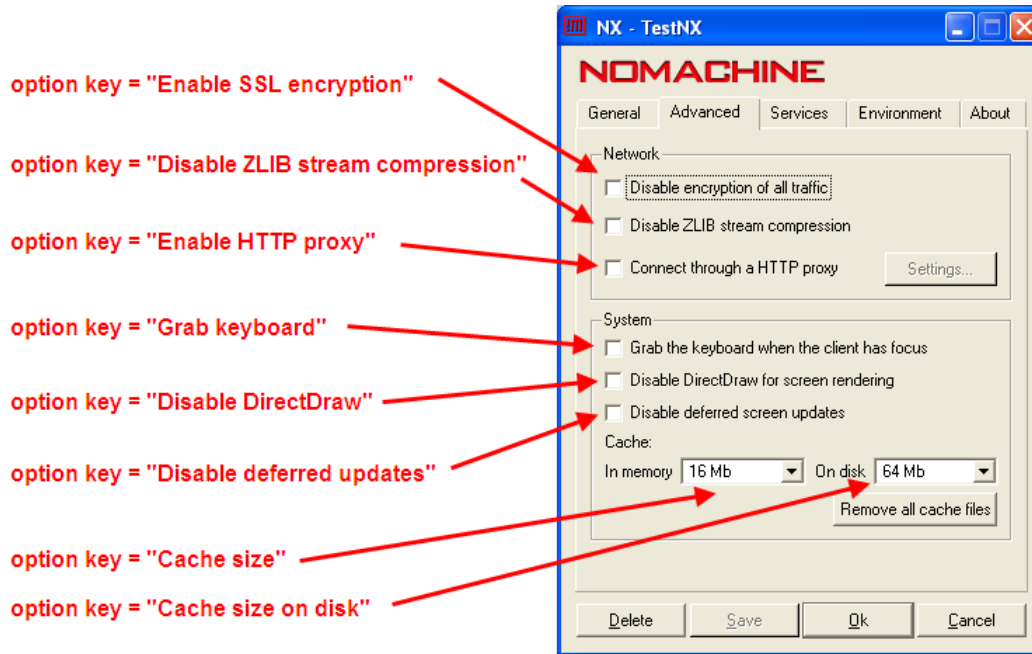


The following table describes the option keys in the `General` group that are most important when integrating with Leostream.

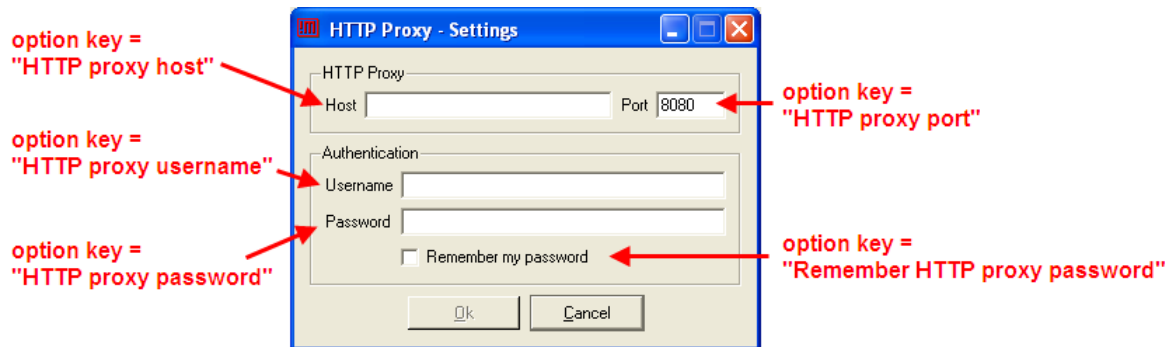
Option Key	Tab / Dialog	Purpose
<code>Desktop</code>	General	(Default = <code>kde</code>) Ensure that the value entered for this option key matches the type of desktop environment used on the remote desktop. For example, typically enter <code>gnome</code> for Ubuntu and <code>kde</code> for Red Hat.
<code>Link speed</code>	General	(Default = <code>lan</code>) Indicate the type of network connection.
<code>Resolution</code>	General	(Default = <code>available</code>) Set the resolution of the opened NX session. If your policy performs disconnect actions, use <code>available</code> to indicate the remote session should fill the available space on the monitor, but still present the NX toolbar at the top. Otherwise, if you set this value to <code>fullscreen</code> , the remote session opens in a seamless window and the user cannot perform a disconnect action. To specify the exact resolution, use the syntax <code>width x height</code> , for example <code>800x600</code>
<code>Resolution height</code>	General	(Default = <code>600</code>) Set the height of the remote session. If a <code>Resolution</code> option key is specified, the NX session ignores the value in <code>Resolution height</code> and uses the <code>Resolution</code> option key to define the complete resolution
<code>Resolution width</code>	General	(Default = <code>800</code>) Set the width of the remote session. If a <code>Resolution</code> option key is specified, the NX session ignores the value in <code>Resolution width</code> and uses the <code>Resolution</code> option key to define the complete resolution.
<code>Server host</code>	General	(Default = <code>{IP}</code>) The IP address or hostname of the remote desktop. The Connection Broker replaces the dynamic tag <code>{IP}</code> with the correct information for the desktop the user selected
<code>Server port</code>	General	(Default = <code>22</code>) The port used for NX.
<code>Session</code>	General	(Default = <code>unix</code>) Use the default value to establish an NX connection to a Linux desktop. The <code>Desktop</code> option key indicates the desktop environment used on the Linux desktop.
<code>Spread over monitors</code>	General	(Default = <code>false</code>) Set to <code>true</code> to specify that the session should span across multiple monitors. For best results when using multiple monitors, set all monitors to the same resolution. Currently, you must hard-code <code>true</code> or <code>false</code> . Connection Broker display plans do not support NX connections.

Advanced Group

The following figures indicate option keys in the configuration file that correspond to fields on the **Advanced** tab in the **Configuration** dialog.



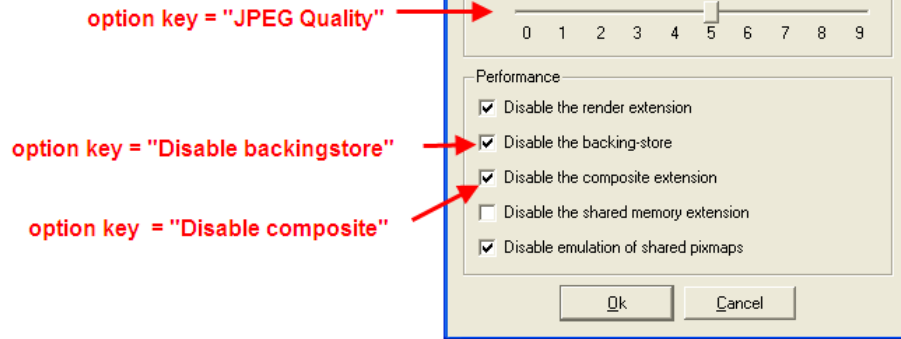
If you select the **Connect through a HTTP proxy** option, the **Settings** button enables. Click the **Settings** button to open the following dialog, which sets the indicated option keys.



Images Group

The following figures indicate option keys in the **Images** group, that apply when launching a Unix session. Other options in the **Images** group apply to VNC, RDP, or other types of sessions, and can be ignored when connecting to the desktop using NX.

The value for the option key "Image Compression Type" is set based on the selected radio button and the state of the "Use custom JPEG quality" option.



Setting User-Configurable Parameters

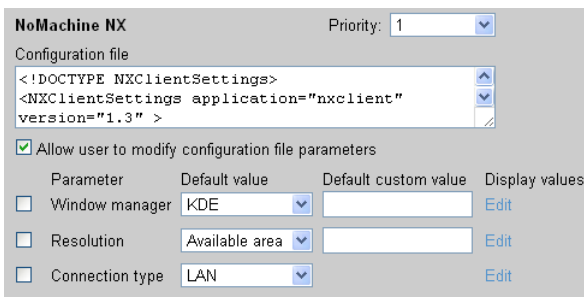
The configuration file in the NoMachine NX section of the protocol plan overrides any parameter settings made on the user's NX software client. As Connection Broker Administrator, you have control over how the user's NX session is launched.

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.

- Window manager
- Resolution
- Connection type

Enabling End-User Configurable Parameters

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



To configure which parameters the user is allowed to modify:

1. Select the checkbox before each parameter (Window manager, Resolution, Connection type) that the user can customize.
2. After selecting which parameters the user can modify, modify the text in the **Configuration file** field to use pre-

defined dynamic tags, which the Connection Broker replaces with the values specified by the default values or by the user. These dynamic tags are, as shown in **bold** in the clip of the configuration file that follows.



If you do not place the dynamic tags in the **Configuration file**, the user-specified settings will not be applied.

Parameter	Option Key	Original Text	Replace with
Window manager	Desktop	value="kde"	value="{WM_TYPE}"
Resolution	Resolution Resolution height Resolution width	value="available" value="600" value="800"	value="{RESOLUTION}" value="{RESOLUTION_HEIGHT}" value="{RESOLUTION_WIDTH}"
Connection type	Link speed	value="lan"	value="{CONNECTION_TYPE}"

3. From the **Default value** drop-down menus, indicate the value to use if the user has not customized the parameter.
4. If you select **Custom** for the default value for window manager or resolution, enter the custom value into the **Default custom value** edit field. For resolution, enter the value as *heightxwidth* where *height* and *width* are in pixels and there is no space between the numbers and the *x*.
5. The drop-down menus in the end-user dialog include all values shown in the **Default value** drop-down menu on the Administrator interface. You can choose to show user-friendly descriptions of these items by defining display values. To define display values:
 - a. Click the **Edit** link in the **Display value** column
 - b. In the **Edit Display Values** form that opens, enter user-friendly names into the **Display value** edit field for each possible internal value.
 - c. Click **Save** on the **Edit Display Values** form. The new display values are shown in the **Default value** drop-down menu, as they will be displayed to users.
6. You must repeat steps 1 through 4 for the **Web browser** and **Leostream Connect** sections of the protocol plan, if users log in from both clients. Both sections use the same display values entered in step 5.

End-User Interface for Configuring Parameters

End-users can set default values for the NX configuration parameters when logging in through the Leostream Web client and Windows version of Leostream Connect.

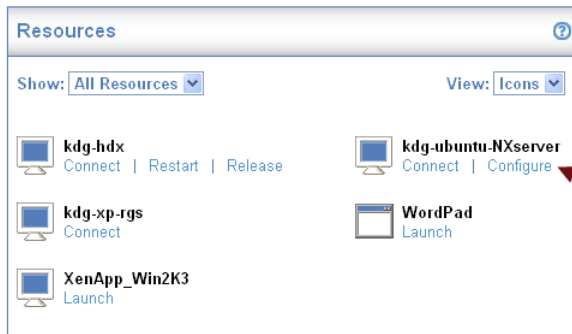


End-user specified parameters are stored on a per client-to-desktop connection. For example, when a user specifies a desired resolution for Desktop A when logging in at Client A, that resolution is used every time the user logs into Desktop A at that client. However, if the user logs into Desktop A from Client B and *does not specify* a desired resolution, the connection is established using the default resolution for the protocol plan. The user must reset their desired resolution for every client connecting to every desktop.

Web Client

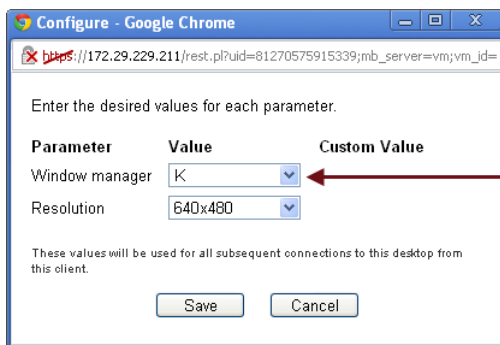
Users logging in from the Leostream Web client set NX parameters, as follows.

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



Click "Configure" set specify custom values for the allowable configuration parameters. Otherwise, default values are used.

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



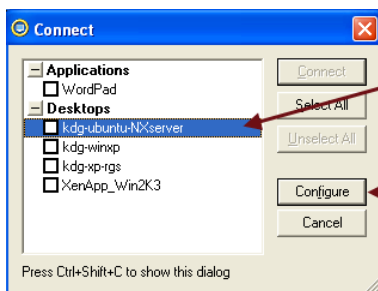
The end-user drop-down menu contains the "Display values" set by the administrator. The specified default value is selected.

3. The user clicks **Save** to store their desired values. Their specified parameters are then used every time the user connects to the configured desktop from that client device. If the user moves to a new client, they must reconfigure the default values for that client.

Leostream Connect

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

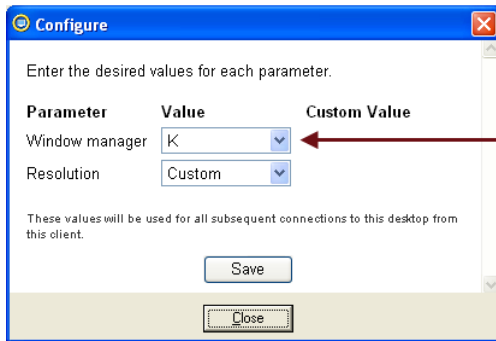
1. After logging into the Leostream Connect client, highlighting any NX connection with configurable parameters enables a **Configure** button, as shown in the following figure.



Highlight any NX connection that has configurable parameters. Checking the box is *not* the same as highlighting.

If the highlighted NX connection has configurable parameters, the "Configure" button enables. Click the button to set desired parameter values.

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



The end-user drop-down menu contains the "Display values" set by the administrator. The specified default value is selected.

3. The user clicks **Save** to store their desired values. Their specified parameters are then used every time the user connects to the configured desktop from that client device. If the user moves to a new client, they must reconfigure the default values for that client.

✓ The Connection Broker considers Leostream Connect and the Leostream Web client accessed from the same physical device as two different clients.

Session Shadowing and Collaboration

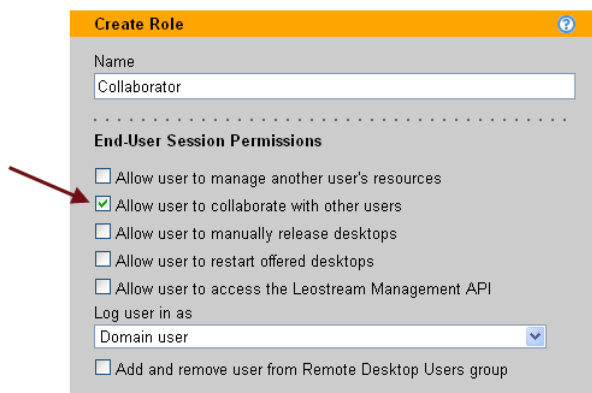
Connection Broker 7.5 allows users that connect to desktops using the NoMachine NX protocol to collaborate by sharing their session or shadowing another user's session.

✓ 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.

- Go to the > **Users > Policies** page.
- Edit the user's policy, or create a new policy.
- In the **Desktop Assignments from Pool** section, select a pool that contains desktops that support collaboration.
- 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

Select this option to enable collaboration for NX sessions established to desktops in this pool. Optionally restrict the shadow session to be non-interactive

Collaboration is supported only for NX sessions.

- By default, shadowed sessions allow the shadow user to interact with the session. To restrict the shadowed session to be view-only, select the **View only shadowing, not interactive** option.
 - 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 interface. The top navigation bar includes 'LEOSTREAM' logo, 'Status', 'Resources', 'Clients', 'Plans', 'Users', 'System', and 'Search'. Below this is a secondary navigation bar with 'Authentication Servers', 'Users', 'Roles', 'Policies', 'Assignments', 'Collaboration', and 'My Options'. The 'Collaboration' page displays a table of invitations with columns for Actions, Type, Status, ID, Session owner, Shadow user, Desktop, Sent, Viewed, Declined, and Date. The table contains three rows of invitation data.

Actions	Type	Status	ID	Session owner	Shadow user	Desktop	Sent	Viewed	Declined	Date
View	Invitation	Cancelled	17	leo	kgondoly	kdg-ubuntu-NXserver	01/27/2012 - 15:36:38			
View Cancel	Invitation	Viewed	18	leo	kgondoly	kdg-ubuntu-NXserver	02/07/2012 - 10:26:46	02/07/2012 - 11:11:04		
View	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	
View Cancel	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.
- **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.

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

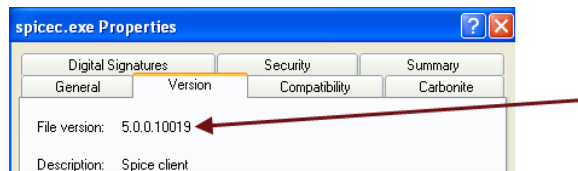
 Leostream Connect does not use the SPICE ActiveX component. You must ensure that `spicec.exe` exists on the client device and that users log in to the Connection Broker using Leostream Connect.


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.

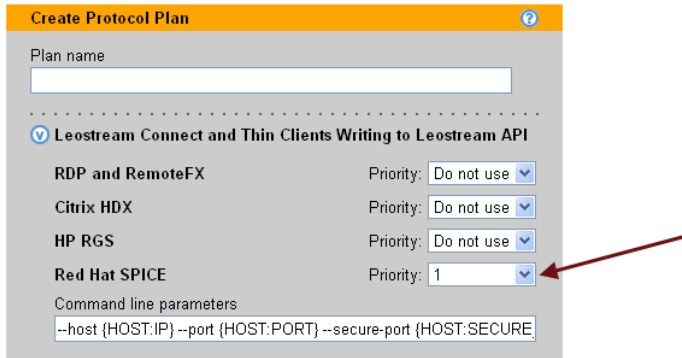


 Older versions of the SPICE client are not compatible with Leostream protocol plans. You must upgrade all `spicec.exe` files to version 5.

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 ticketed 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.

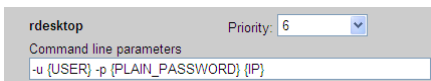
7. Save the protocol plan.

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. By default, rdesktop has a **Priority** of 6. 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.

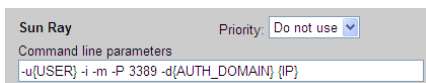
You can use any `rdesktop` command line option, such as `-f` for full screen mode. See the `rdesktop` [documentation](#) for a description of supported command line parameters.

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 – ttatsc

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` parameters 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.

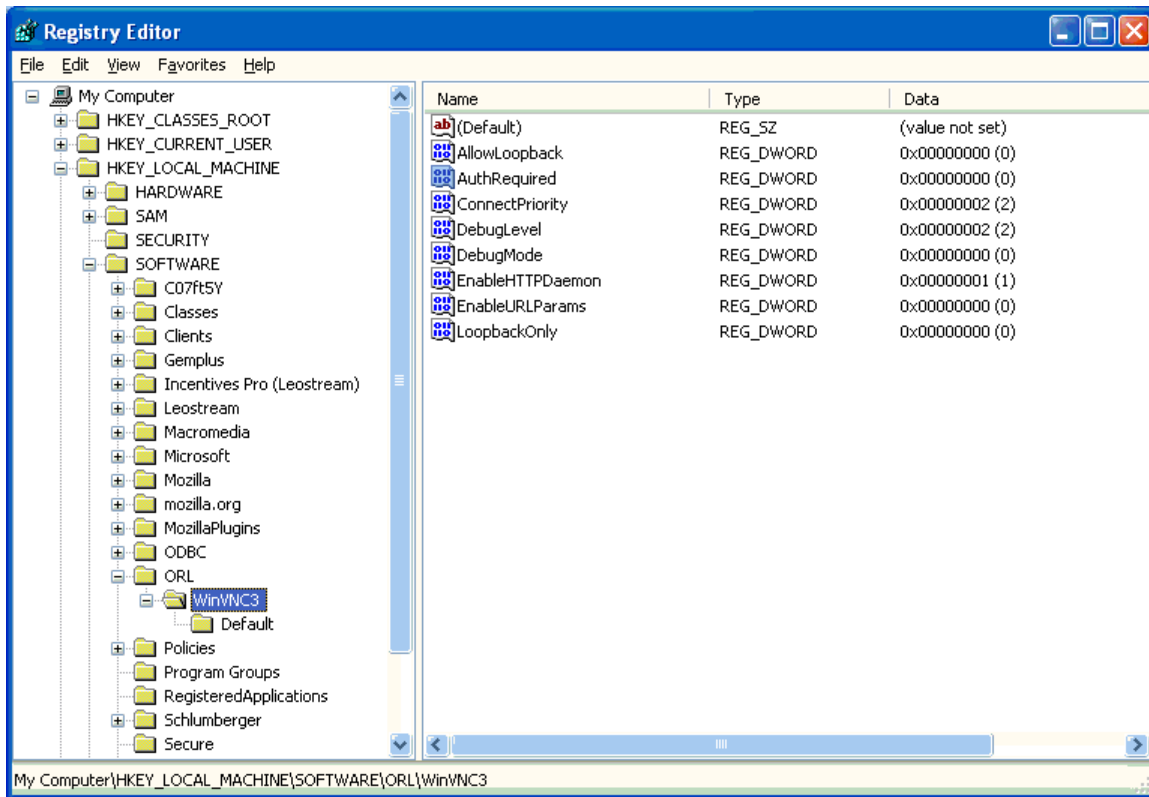


Single sign-on is support only for UltraVNC. To support single sign-on with VNC, you must select the single sign on task when installing the Leostream Agent on the remote desktops. Also, these users must have the **Enable single-sign-on to desktop console** option selected in the **When User is Assigned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

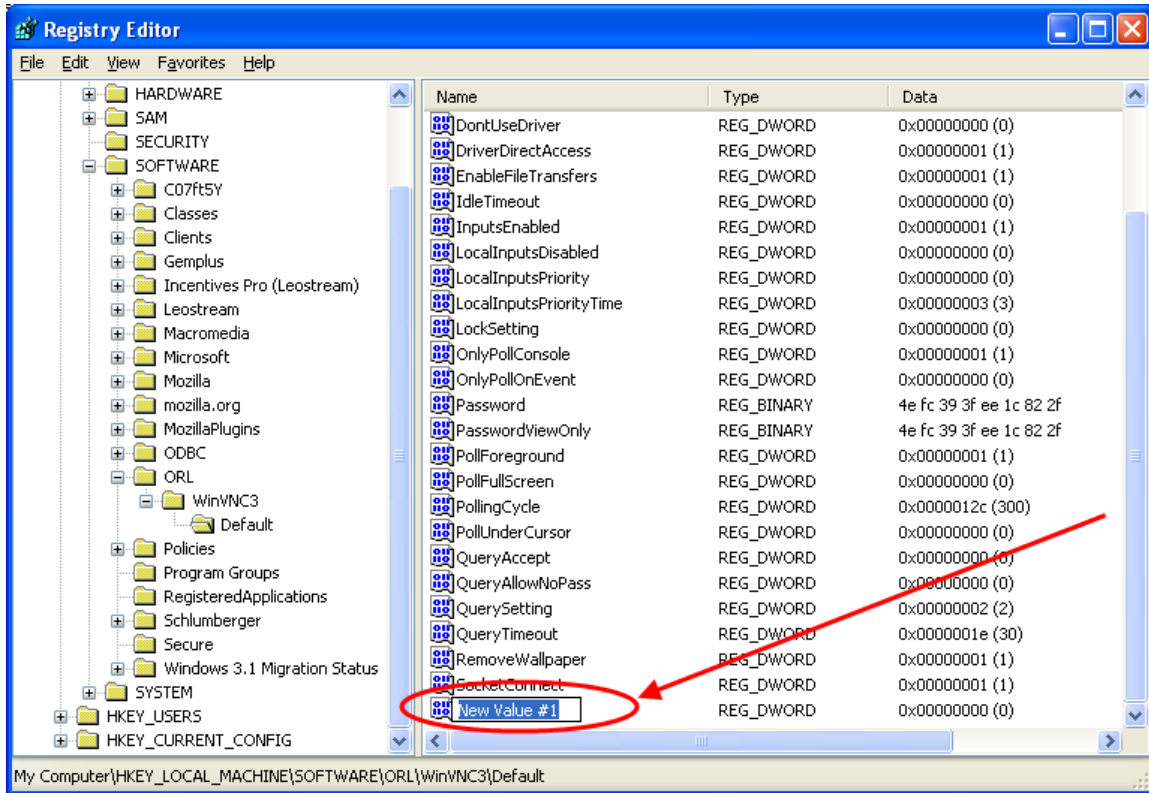
Setting up VNC for Single Sign-On on Windows Operating Systems

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.

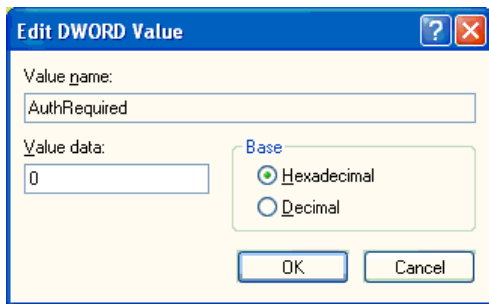
1. Run the `regedit` command to open the **Registry Editor** dialog.
2. Navigate to the `HKEY_LOCAL_MACHINE/SOFTWARE` folder.
3. Inside this folder, open the folder for your VNC installation, for example, `ORL/WinVNC3`, as shown in the following figure.



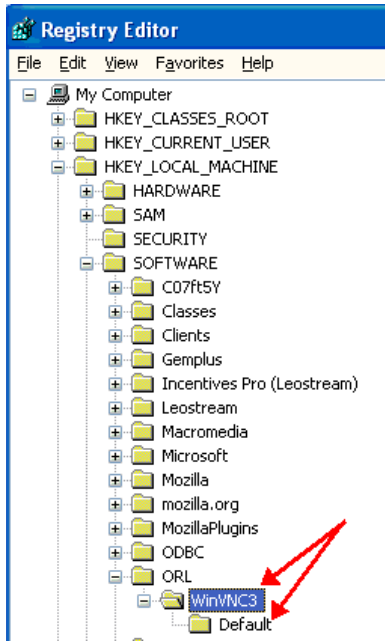
4. Select the `AuthRequired` key. If this key does not exist, create the key as follows:
 - a. Right-click in the list on the right side of the **Registry Editor** and select **New**.
 - b. In the menu that opens, select **DWORD Value**. A new key appears in the list, as shown in the following figure.



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

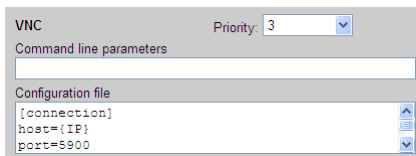


Restart the VNC service after you have set all the keys. The VNC server now accepts a null password. To set the password to null, go to the **Administrator Properties** for the VNC server and empty out the password field.

Setting up the Connection Broker to Use VNC

To configure a protocol plan to use VNC

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



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



The 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® 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
```