UDS Installation, Administration and User Manual

UDS 1.5 Rev. 6 September 1st 2014

CONTENT

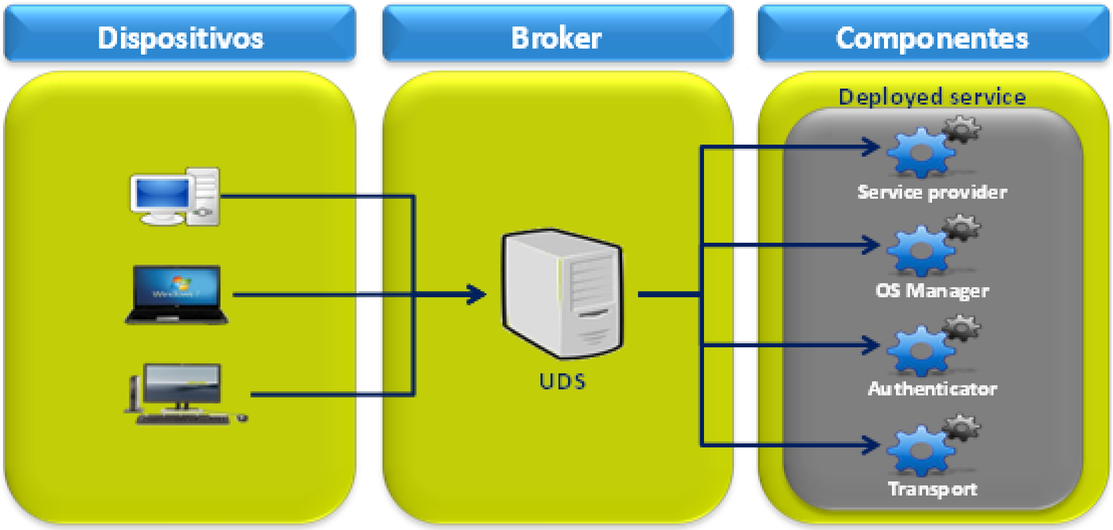
1 INTRODUCTION

UDS (Universal Desktop Services) is a multiplatform VDI connection broker for Windows and Linux. It manages virtual desktop lifecycle and access for virtual desktop platforms and physical resources in the Data Center or Cloud.

UDS provides a set of software elements that forms a platform for managing and deploying IP services.

One of these services is the lifecycle management, administration and connection to virtual desktops.

This document contains basic instructions for installing UDS software on a virtual infrastructure and the administration and management of the service aimed at virtual desktop platforms.



1.1 Features

The main features of UDS include:

- Very easy installation and administration.
- Multi-hypervisor, with the ability to migrate the platform to more efficient future solutions (currently vSPhere and KVM, and Hyper-V is in development).
- Multi-authenticator, which permits users and user groups from different sources to be set up with a practically unlimited number of configurations.
- Authentication system via multiple connectors, for example: Active Directory, eDirectory, OpenLDAP, SAML, LDAP….
- Secure WAN access for publishing PCs on the Internet, using an SSL tunneler included in the subscription.
- Client tool personalization using customized development.
- Product road map based on client and community requests.
- Ready for heterogeneous environments where other solutions do not have access because of functionalities or cost scaling, for example: AAPP or academic environment.
- Subscription model based on support and updates for the implemented platform (UDS Broker).
- Non-redistributable subscription model by segments to an unlimited number of virtual desktops

1.2 VDI platform architecture with UDS

An optimal design of a VDI platform is essential in order to obtain all the benefits which may be provided by the architecture. Each layer that forms a VDI architecture may be designed to fulfill its function without affecting the other ones.

The main elements that form a VDI architecture with UDS are:

- Connection clients: Devices used to access the virtual desktops, such as thin clients, zero clients, PCs, etc…. It is important to identify if the access to the virtual desktops will be carried out from a LAN or from WAN.

- UDS Servers: They are formed by a DB to storage all the data related to the environment, a connection broker which will manage the virtual desktops' lifecycle and communication with the hypervisors and a tunnel server to allow secure access from outside. All of them will be served in virtual appliance format.

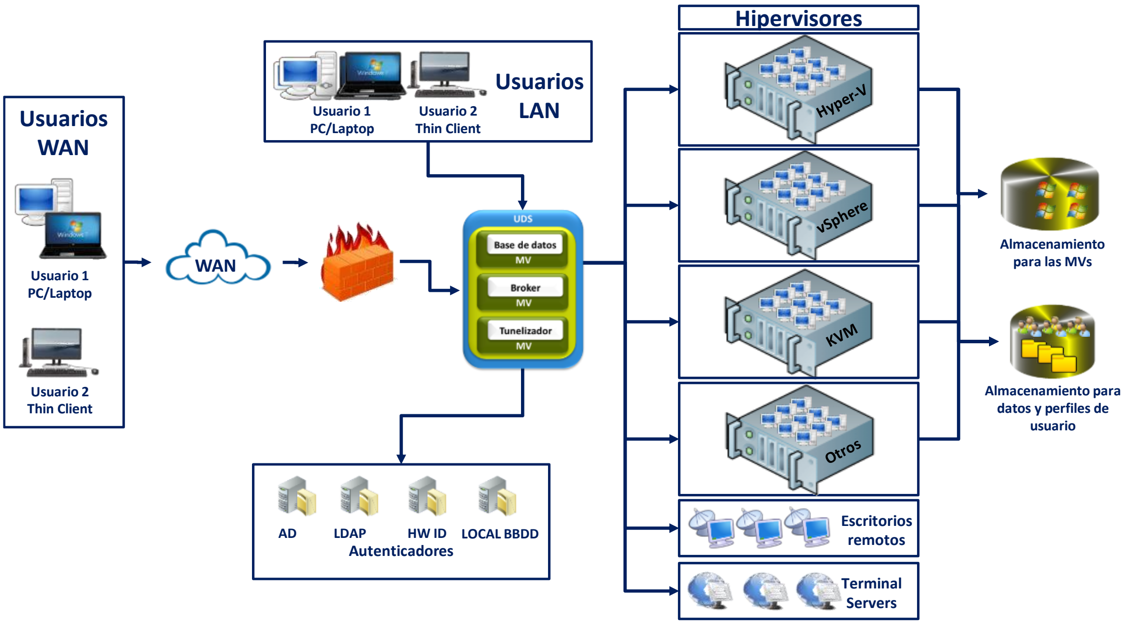
- Authenticator/s: Active Directory, OpenLDAP, eDirectory Servers, etc… Through their integration with UDS they will control the users access to virtual desktops. Depending the environment, you may have from one to an unlimited number of authenticators.

- Hypervisor platform: It executes the creation, switch on and removal of the virtual desktops which are managed from the broker. UDS integrates itself with Microsoft Hyper-V, VMware vSphere and KVM (oVirt and Red Hat Enterprise Virtualization) hypervisors.

- Storage: They will host the servers and/or virtual desktops of the platform. The choice of the type of storage is an important part of the design. Depending on the needs demanded by the users in the virtual desktops, we may select the most appropriate one regarding performance.

With a clear idea of the architecture design, you may start scaling the platform, bearing in mind the number of users that will access to it.

In the following image you can see an example of a VDI architecture with UDS:

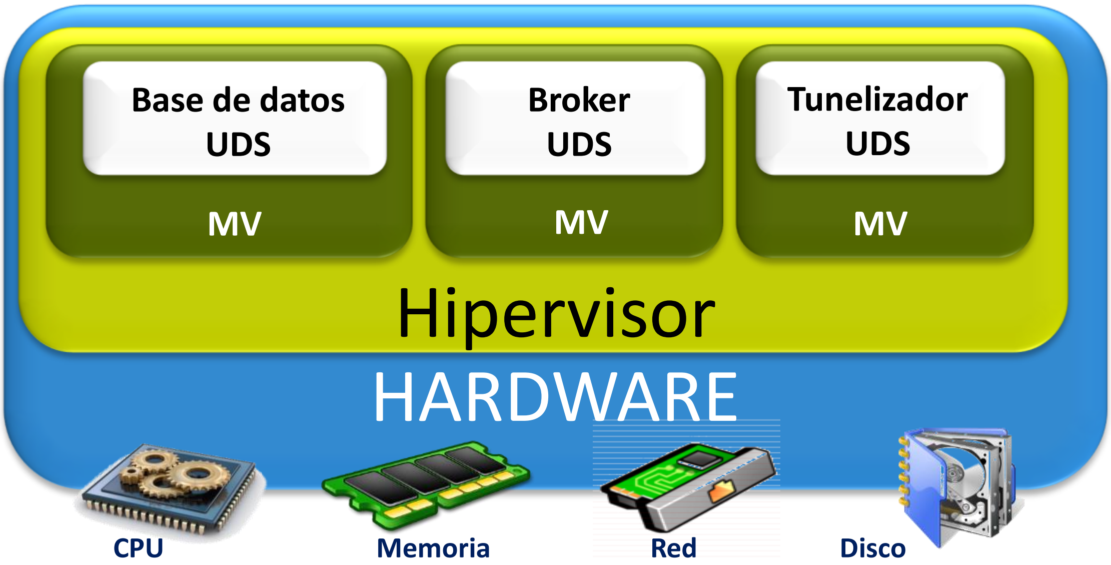


1.3 UDS Components

UDS is made up of 5 elements that interact with each other.

- UDS Broker, UDS Tunneler and UDS Database: each of these 3 elements are installed on a virtual machine (VM) and they are provided in virtual appliance format.
- UDS Actor: it is installed on the VM as a service that will be used as a template for deploying the desktop groups.
- UDS Administration Client: it is installed on any of the supported OS and can be hosted on a physical or virtual machine.

NOTE: We strongly recommend to replace the administration client by the new GUI for web administration (see section 4 "ADMINISTERING UDS - WEB ADMINISTRATION")



The features and technical requirements of each component are defined below:

1.3.1 UDS Broker

This is the software that mediates between clients and service providers.

This is the basic piece of UDS, as it performs the functions of connection broker to the desktops and permits the administration and management of virtual desktop platforms defined as implemented services.

Virtual Appliance with the following features:

- Virtual hard drive: 3 GB.
- Memory: 768 MB.
- CPU: 2 vCPU.
- Network: 1 vNIC.

Requirements:

- 1 IP Direction.
- IP DNS.
- Network mask.
- IP Gateway.
- Domain name.
- Database IP.
- DB port and instance name.
- Activation code.

1.3.2 UDS Tunneler

Software that establishes secure connections to virtual desktops through WAN. It also provides HTML5 access to the virtual desktops.

UDS tunneler allows the connection from any device/browser/client to the virtual desktops through a SSH tunnel without having installed any software beforehand. Moreover, it allows RDP access to virtual desktops through HTML5.

Virtual Appliance with the following features:

- Hard drive: 3 GB .
- Memory: 768 MB.
- CPU: 2 vCPU.
- Network: 1 vNIC.

Requirements:

- 1 IP Direction.
- IP DNS.
- Network mask.
- IP Gateway.
- Domain name.
- IP Broker.

1.3.3 UDS Database

This component is responsible for storing all system UDS data, such as service providers, authenticators, connectivity… and all the information needed to generate statistics.

Currently, in UDS 1.5 version, only the database manager MySQL version 5.x is supported.

It is necessary to have an appropriately configured MySQL database with a valid instance and user at the time of installation.

IMPORTANT!!!: In the event that you do not have said database manager, VirtualCable can provide this component as a Virtual Appliance. This component is not included in the UDS support.

Virtual Appliance with the following characteristics:

- Hard drive: 2 GB.
- Memory: 1 GB
- CPU: 1 vCPU.
- Network: 1 vNIC.

Requirements:

- 1 IP address.
- DNS IP.
- Network mask.
- IP Gateway.
- Domain name.
- DB instance name.
- User with instance permission.

1.3.4 UDS Actor

This software performs the communication and interface functions for transmitting data (virtual desktop status, machine name…) and commands between the broker and the virtual desktops managed by UDS.

It is installed on the virtual machine as a service that will be used as a template for generating virtual desktop groups based on linked clones.

- The supported operating systems are:
- Windows 8.
- Windows 7.
- Windows XP.
- Windows 2003.
- Windows 2008.
- Linux (distributions based on Debian).

Requirements:

- .Net Framework 3.5 SP1 (Windows machines).
- Python 2.6 or higher (Linux machines).
- UDS broker machine IP.

1.3.5 UDS Administration client

This software is responsible for providing access to the UDS administration and management interface.

The supported operating systems are:

- Windows 7.
- Windows XP.
- Windows 2003.
- Windows 2008.

Requirements:

- UDS Broker IP.
- User with administration rights on the UDS Broker.
- .Net Framework 3.5 SP1.

NOTE: We strongly recommend to replace the administration client by the new GUI for web administration (see section 4 "ADMINISTERING UDS - WEB ADMINISTRATION")

2 BEFORE INSTALLING UDS

The UDS components can be hosted on different virtualization platforms.

Even though the UDS components are hosted on a single virtual platform, UDS is capable of managing the deployment of virtual desktops on multiple virtual platforms that are completely independent of the virtual platform where UDS is hosted.

The content of this section describes the requirements for installing UDS on different virtualization platforms and the requirements of the virtual platform on which the software is to be installed.

2.1 Installation of UDS on VMware vSphere virtual platform

2.1.1 Virtual platform requirements

UDS will be able to be deployed on VMware vSphere platforms starting with version 5.x.

To find out the requirements of a VMware vSphere platform, you can access the following documentation:

- [VMware Compatibility Guide]
- [vCenter Server and vSphere Client Hardware Requirements]

The VMware platform on which UDS will be deployed must meet the following requirements:

- At least one VMware ESXi server with a valid license is needed for hosting the UDS servers and generating the virtual desktops.
- The vSphere platform must be administered by a vCenter with a valid license.

For UDS to be installed and capable of sending requests to a vCenter, and for these requests to be carried out, the user must have administration rights credentials on the VMware vSphere platform on which the virtual desktops are to be deployed.

- At least one Virtual Machine Port Group to which the Virtual Appliance of the UDS platform is going to be connected must be established.
- At least one Virtual Machine Port Group to which the different virtual desktops managed by UDS are going to be connected must be established.
- There must be at least 8 GB of free space on the hard drive to host the Virtual Appliance that make up UDS.
- There must be at least 8 GB of free RAM to host the Virtual Appliance that make up UDS.

2.1.2 Network connections

The following connections between the different elements which make up the UDS platform must be enabled:

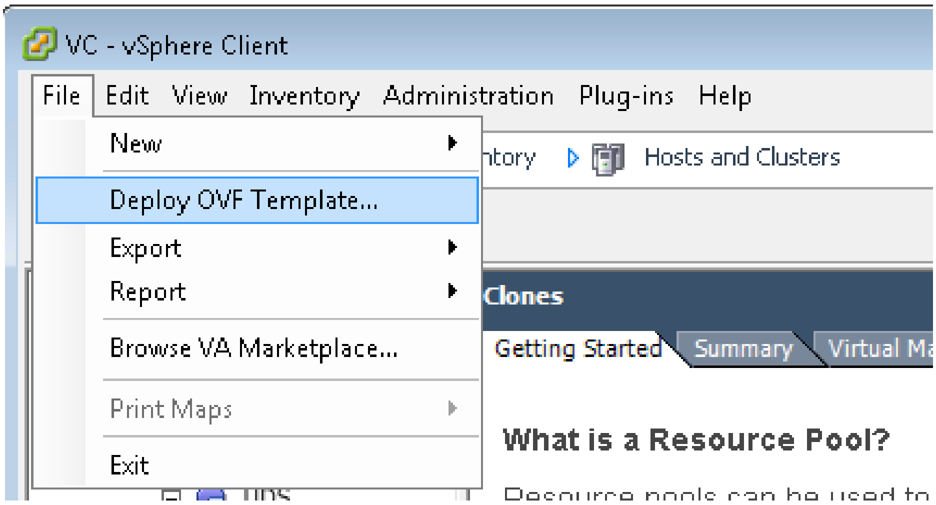
Origin	Destination	Port
UDS Broker	DB MySQl	3306
UDS Broker	vCenter	443
UDS Broker	Authenticator	389, 636 (SSL)
Tunneler	UDS Broker	80, 443
UDS Broker	Virtual desktops	3389 (RDP), 22 (NX), 42966 (RGS)
Tunneler	Virtual desktops	3389 (RDP), 22 (NX), 42966 (RGS)
Users	UDS Broker	80, 443
Users	Tunneler	443
Users	HTML5 (Tunnel)	10443

2.1.3 Storing UDS on the VMware vSphere platform

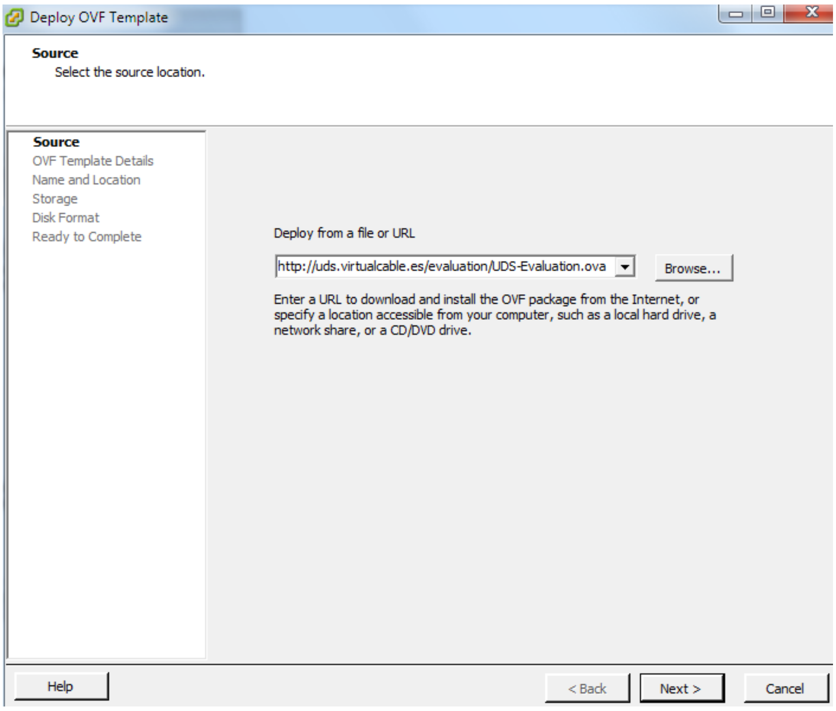
The main component of UDS evaluation release is provided in OVA format (Open Virtualization Alliance) to upload the VM to vSphere platform, using the vSphere client.

Steps:

1.- Once established connection on the target platform though VMware vSphere client, please choose the menu option **File \ deploy OVF Template ...**



2.- Select the source location of the virtual machine .ova file or, if you have Internet access and credentials were provided to you when you subscribed to UDS Enterprise, directly enter the download address: http://uds.virtualcable.es/enterprise/UDS-Server.ova



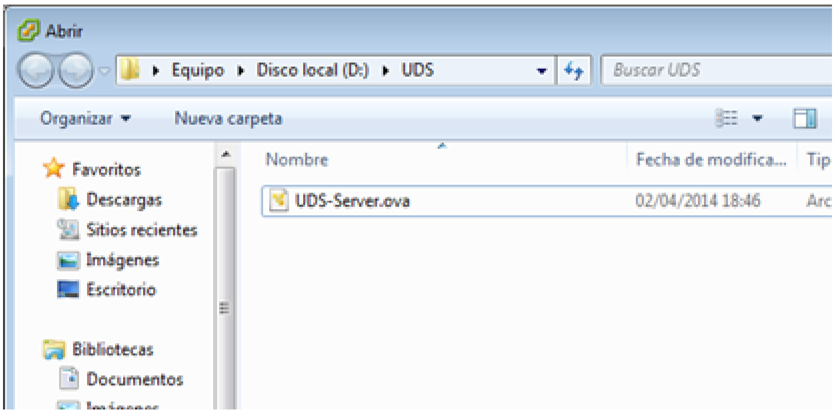
Deploy OVF Template

Source

Select the source location.

Deploy from a file or URL: http://uds.virtualcable.es/evaluation/UDS-Evaluation.ova

Enter a URL to download and install the OVF package from the Internet, or specify a location accessible from your computer, such as a local hard drive, a network share, or a CD/DVD drive.

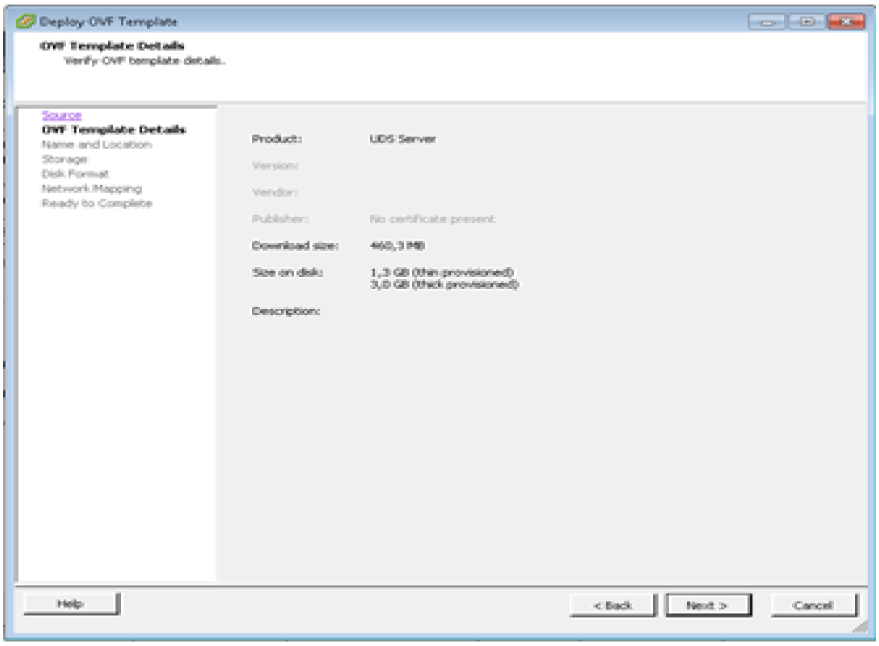


Abrir

Equipo ▸ Disco local (D:) ▸ UDS

Nombre	Fecha de modifica...	Tip
UDS-Server.ova	02/04/2014 18:46	Arc

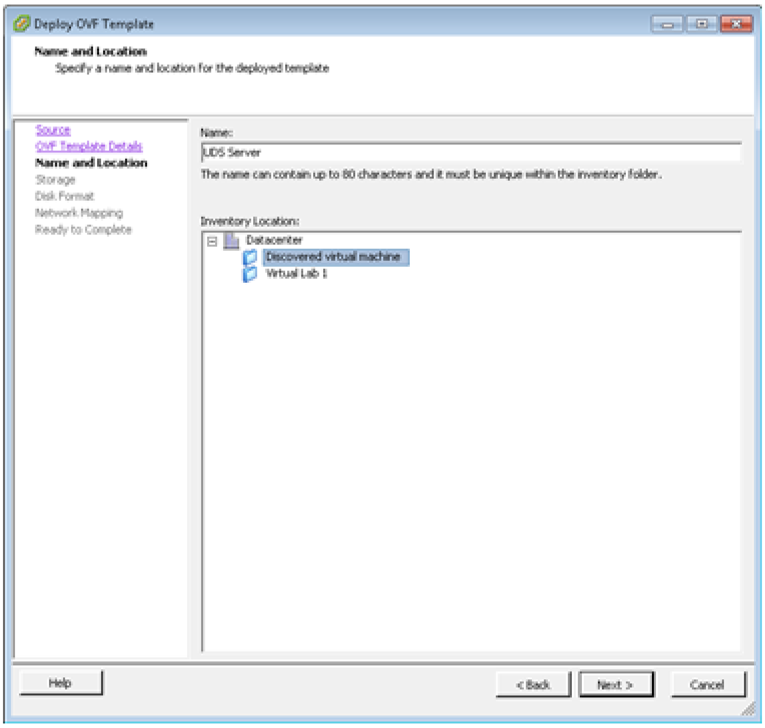
3.- On next step the wizard displays virtual machine features to be hosted in the target virtual platform:



OVF Template Details
Verify OVF template details.

Product: UDS Server
Version:
Vendor:
Publisher: No certificate present
Download size: 460,3 MB
Size on disk: 1,3 GB (thin provisioned)
3,0 GB (thick provisioned)
Description:

4.- Next step, select inventory name and location of the virtual machine on the target platform.



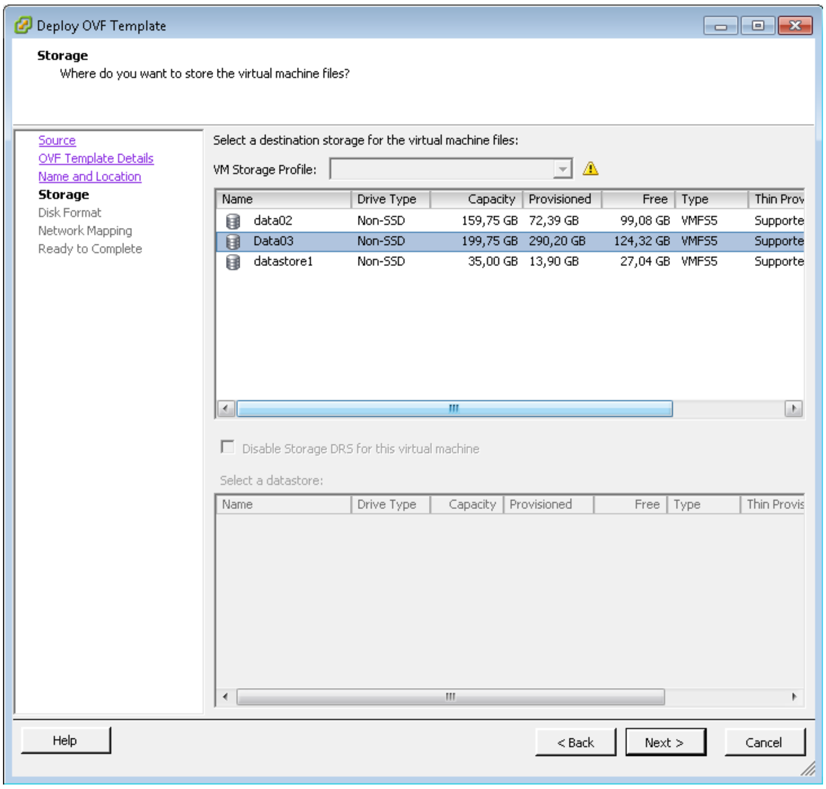
Name and Location
Specify a name and location for the deployed template

Name: UDS Server

The name can contain up to 80 characters and it must be unique within the inventory folder.

Inventory Location:

5.- Then, select the destination datastore where the virtual machine is going to be stored.



Deploy OVF Template

Storage

Where do you want to store the virtual machine files?

Select a destination storage for the virtual machine files:

VM Storage Profile:

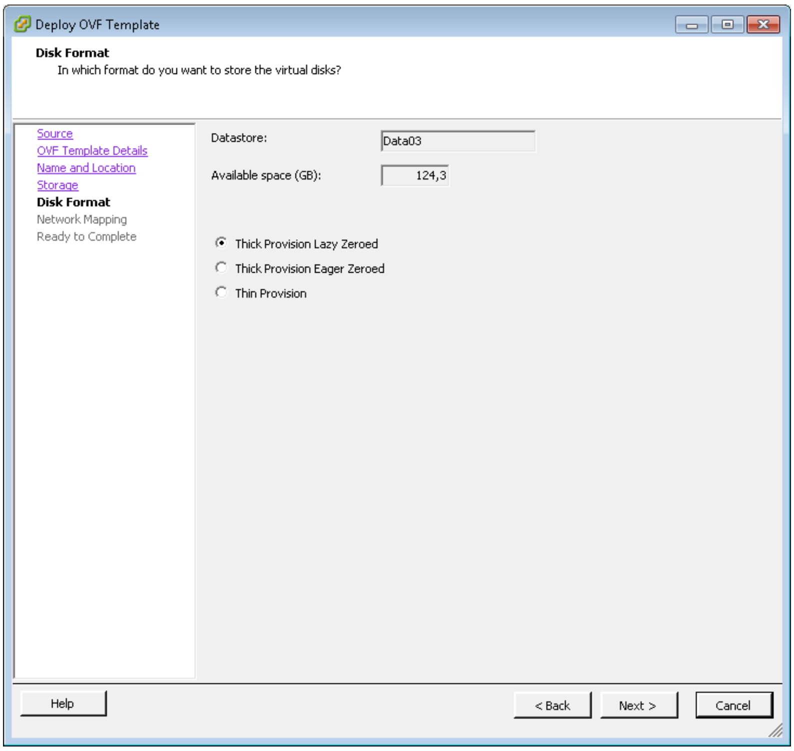
Name	Drive Type	Capacity	Provisioned	Free	Type	Thin Prov
data02	Non-SSD	159,75 GB	72,39 GB	99,08 GB	VMFS5	Supporte
Data03	Non-SSD	199,75 GB	290,20 GB	124,32 GB	VMFS5	Supporte
datastore1	Non-SSD	35,00 GB	13,90 GB	27,04 GB	VMFS5	Supporte

□ Disable Storage DRS for this virtual machine

Select a datastore:

Name	Drive Type	Capacity	Provisioned	Free	Type	Thin Provis

6.- Next step, the wizard displays the name and size of the selected datastore; you can choose the machine virtual hard drive format. It is recommended to select **"Thick Provision Lacy Zeroed"** format, as this provides better performance.



Deploy OVF Template

Disk Format

In which format do you want to store the virtual disks?

Datastore: Data03

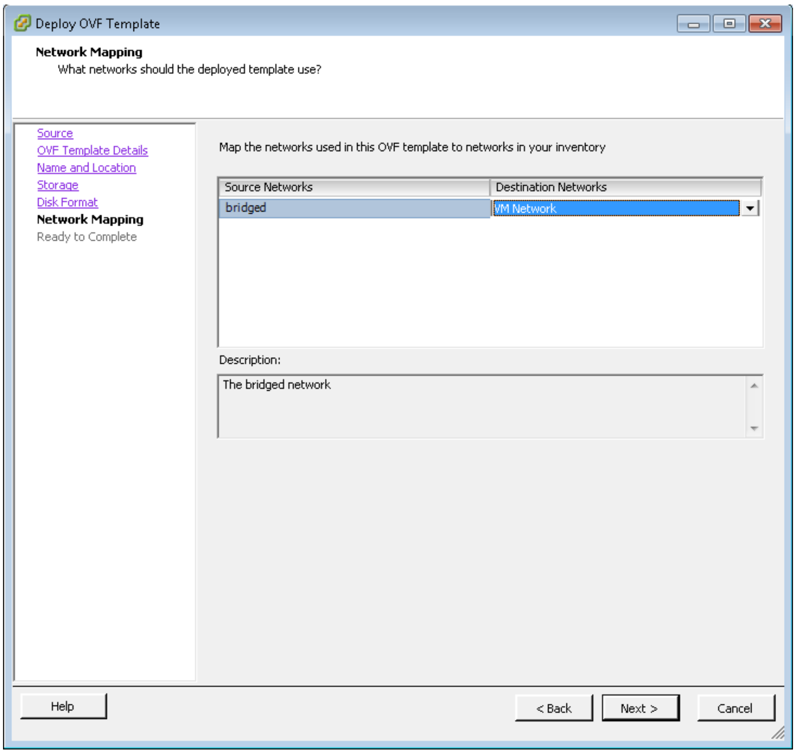
Available space (GB): 124,3

● Thick Provision Lazy Zeroed

○ Thick Provision Eager Zeroed

○ Thin Provision

7.- Select the virtual network to connect the virtual machine.



Network Mapping

What networks should the deployed template use?

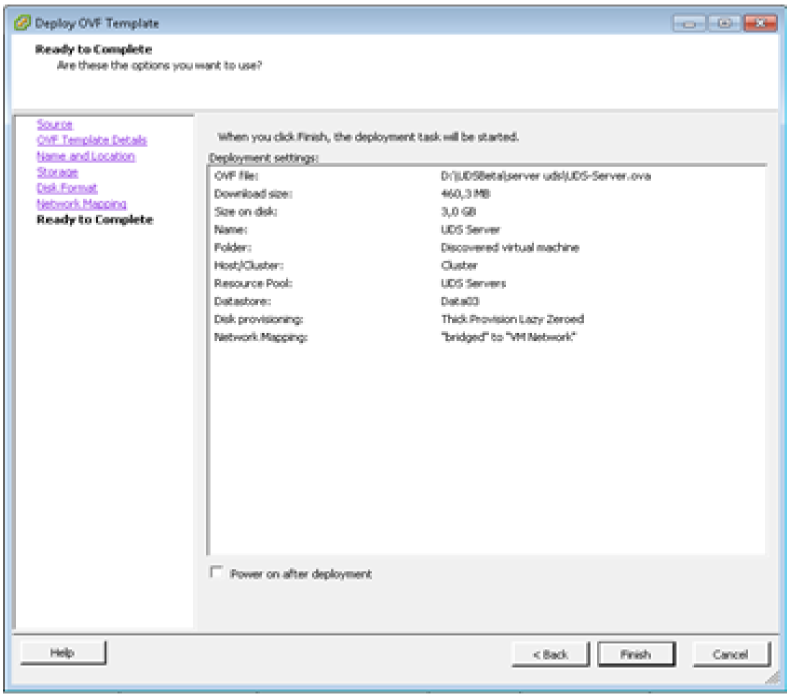
Map the networks used in this OVF template to networks in your inventory

Source Networks	Destination Networks
bridged	VM Network

Description:

The bridged network

8.- In the last step, wizard displays the virtual machine conversion details. Clicking "Finish" button starts the conversion process.



Ready to Complete

Are these the options you want to use?

When you click Finish, the deployment task will be started.

Deployment settings:

OVF file:	D:\UDSBeta\server uds\UDS-Server.ova
Download size:	460,3 MB
Size on disk:	3,0 GB
Name:	UDS Server
Folder:	Discovered virtual machine
Host/Cluster:	Cluster
Resource Pool:	UDS Servers
Datastore:	Data03
Disk provisioning:	Thick Provision Lazy Zeroed
Network Mapping:	"bridged" to "VM Network"

□ Power on after deployment

Once the conversion process has been completed, you now have the UDS broker in its Enterprise version stored on the vSphere virtual platform.

NOTE: Steps 1 to 8 (if necessary) must be repeated for the Virtual Appliance of the tunneler and the MySQL Database server (only in the event that Virtual Cable provides the Virtual Appliance database).

The tunneler download link is:

http://uds.virtualcable.es/enterprise/Tuneler.ova

The MySQL server download link is:

http://uds.virtualcable.es/enterprise/Mysql.ova

2.2 UDS installation on oVirt virtual platform

2.2.1 Virtual platform requirements

UDS can be deployed on oVirt platforms starting with version 3.2.

The oVirt platform on which UDS is going to be deployed must meet the following requirements.

- At least one oVirt server node is needed to host the UDS servers and create the virtual desktops.
- The oVirt platform must be administered by an oVirt-engine.

For UDS to be installed and capable of sending requests to an oVirt-engine, and for these requests to be carried out, the user must have administration rights credentials on the oVirt platform on which the virtual desktops are to be deployed.

- You must have at least one cluster set up for creating and configuring the different virtual desktops managed by UDS.
- You must have at least one “Logical network” set up to which the virtual servers of the UDS platform are going to be connected.
- You must have at least one “Logical network” set up to which the different virtual desktops managed by UDS are going to be connected.
- There must be at least 8 GB of free space on the hard drive to host the virtual servers that make up UDS.
- There must be at least 4 GB of free RAM to host the virtual servers that make up UDS.

2.2.2 Network connections

The following ports between the different elements that make up the UDS platform must be enabled:

Origin	Destination	Port
UDS Broker	DB MySQL	3306
UDS Broker	oVirt-engine	443
UDS Broker	Authenticator	389, 636 (SSL)
Tunneler	UDS Broker	80, 443
UDS Broker	Virtual desktops	3389 (RDP), 22 (NX), 42966 (RGS)
Tunneler	Virtual desktops	3389 (RDP), 22 (NX), 42966 (RGS)
Users	UDS Broker	80, 443
Users	Tunneler	443
Users	HTML5 (Tunnel)	10443

2.3 UDS installation on RHEV virtual platform

2.3.1 Virtual platform requirements

UDS can be deployed on Red Hat Enterprise Virtualization platforms version 3.

The RHEV platform on which UDS is going to be deployed must meet the following requirements:

- At least one RHEV server is needed to host the UDS servers and create the virtual desktops.
- The RHEV platform must be administered by a RHEV-Manager server.

For UDS to be installed and capable of sending requests to a RHEV-Manager, and for these requests to be carried out, the user must have administration rights credentials on the RHEV platform on which the virtual desktops are to be deployed.

- You must have at least one set up cluster for creating and configuring the different virtual desktops managed by UDS.
- You must have at least one set up “Logical network” to which the virtual servers of the UDS platform are going to be connected.
- You must have at least one set up “Logical network” to which the different virtual desktops managed by UDS are going to be connected.
- There must be at least 8 GB of free space on the hard drive to host the virtual servers that make up UDS.
- There must be at least 4 GB of free RAM to host the virtual servers that make up UDS.

2.3.2 Network connections

The following ports between the different elements that make up the UDS platform must be enabled:

Origin	Destiny	Port
UDS Broker	DB MySQL	3306
UDS Broker	RHEV-Manager	80, 443
UDS Broker	Authenticator	389, 636 (SSL)
Tunneler	UDS Broker	80, 443
UDS Broker	Virtual Desktops	3389 (RDP), 22 (NX), 42966 (RGS)
Tunneler	Virtual Desktops	3389 (RDP), 22 (NX), 42966 (RGS)
Users	UDS Broker	80, 443
Users	Tunneler	443
Users	HTML5 (Tunnel)	10443

2.4 UDS installation on Microsoft Hyper-V (experimental)

2.4.1 Virtual platform requirements

UDS can be deployed on Microsoft Hyper-V platforms starting with version 3

The Microsoft Hyper-V platform on which UDS is going to be deployed must meet the following requirements.

- At least one Microsoft Hyper-V server with a valid license to host the UDS servers and create the virtual desktops.
- It is necessary that Microsoft Hyper-V servers are not part of a Microsoft cluster.

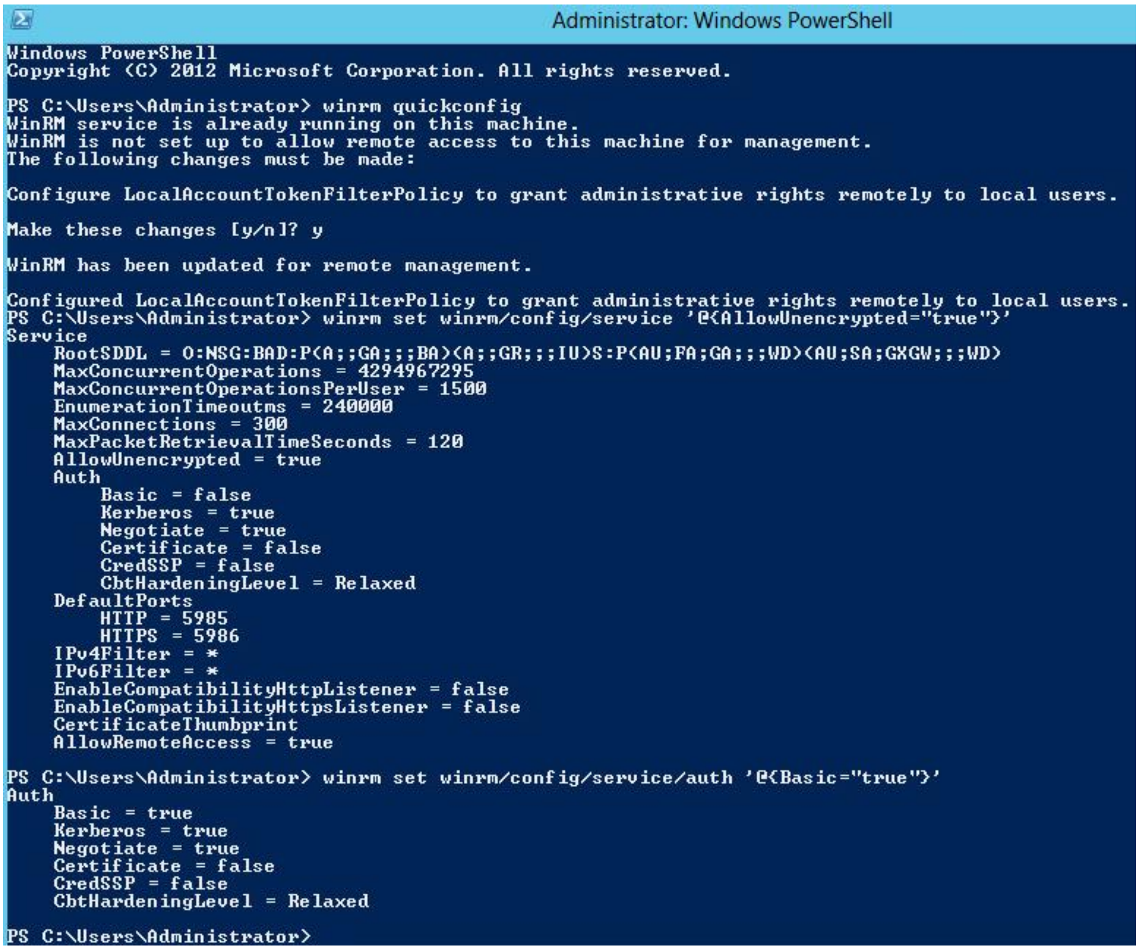
For UDS to work properly against a Microsoft Hyper-V server, it is necessary that this server is not part of a Microsoft cluster. The support for clustered Microsoft Hyper-V will be supported in next UDS versions.

- You must have at least one Virtual Switch to connect the virtual servers of UDS platform.
- You must have at least one Virtual Switch to connect the different virtual desktops managed by UDS.
- You must have the credentials of one user with administration privileges on the Microsoft Hyper-V platform where the virtual desktops are going to be deployed.
- There must be at least 8 GB of free space on the hard drive to host the virtual servers that make up UDS.
- There must be at least 4 GB of free RAM to host the virtual servers that make up UDS.
- You must enable WSMan on every Hyper-V host used with UDS so that Microsoft Hyper-V with UDS will perform properly.

To enable it through HTTPS, you must have a valid certificate.

To enable it through HTTP, we run:

- winrm quickconfig
- winrm set winrm/config/service '@{AllowUnencrypted="true"}'
- winrm set winrm/config/service/auth '@{Basic="true"}'



```
Windows PowerShell
Copyright (C) 2012 Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> winrm quickconfig
WinRM service is already running on this machine.
WinRM is not set up to allow remote access to this machine for management.
The following changes must be made:

Configure LocalAccountTokenFilterPolicy to grant administrative rights remotely to local users.

Make these changes [y/n]? y

WinRM has been updated for remote management.

Configured LocalAccountTokenFilterPolicy to grant administrative rights remotely to local users.
PS C:\Users\Administrator> winrm set winrm/config/service '@{AllowUnencrypted="true"}'
Service
    RootSDDL = O:NSG:BAD:P(A;;GA;;;BA)(A;;GR;;;IU)S:P(AU;FA;GA;;;WD)(AU;SA;GXGW;;;WD)
    MaxConcurrentOperations = 4294967295
    MaxConcurrentOperationsPerUser = 1500
    EnumerationTimeoutms = 240000
    MaxConnections = 300
    MaxPacketRetrievalTimeSeconds = 120
    AllowUnencrypted = true
    Auth
        Basic = false
        Kerberos = true
        Negotiate = true
        Certificate = false
        CredSSP = false
        CbtHardeningLevel = Relaxed
    DefaultPorts
        HTTP = 5985
        HTTPS = 5986
    IPv4Filter = *
    IPv6Filter = *
    EnableCompatibilityHttpListener = false
    EnableCompatibilityHttpsListener = false
    CertificateThumbprint
    AllowRemoteAccess = true

PS C:\Users\Administrator> winrm set winrm/config/service/auth '@{Basic="true"}'
Auth
    Basic = true
    Kerberos = true
    Negotiate = true
    Certificate = false
    CredSSP = false
    CbtHardeningLevel = Relaxed

PS C:\Users\Administrator>
```

2.4.2 Network connections

The following connections between the different elements which make up the UDS platform must be enabled:

Origin	Destination	Port
UDS Broker	DB MySQL	3306
UDS Broker	Hyper-V	80, 443
UDS Broker	Authenticator	389, 636 (SSL)
Tunneler	UDS Broker	80, 443
UDS Broker	Virtual desktops	3389 (RDP), 22 (NX), 42966 (RGS)
Tunneler	Virtual desktops	3389 (RDP), 22 (NX), 42966 (RGS)
Users	UDS Broker	80, 443
Users	Tunneler	443
Users	HTML5 (Tunnel)	10443

3 INSTALLING UDS

At this point, we will detail the installation of UDS components. The installation procedure will be the same for the different virtualization platforms supported by UDS.

3.1 UDS platform requirements

3.1.1 Infrastructure requirements

The infrastructure requirements needed to be able to deploy UDS are:

- **Virtualization platform.** This will be responsible for hosting the virtual desktops generated by UDS and running the servers that make up UDS.
    - Username and Password of the manager of the virtualization platform with administrator permission.

- **DNS server.** This service is necessary for both the proper operation of the virtual platform as well as for the UDS platform to be deployed.

- **DHCP server.** A DHCP server that permits you to assign IP addresses to the virtual desktop groups created by Linked Clones is needed.

3.1.2 Network requirements

In order to configure the UDS network, you must have at least 3 IP addresses to configure UDS (broker, tunneler and MySQL Database).

It is also necessary to have the following components available:

- Network mask.
- IP address of the DNS server.
- Gateway IP address.
- Domain name (if there is one).
- IP address of the virtualization platform manager.

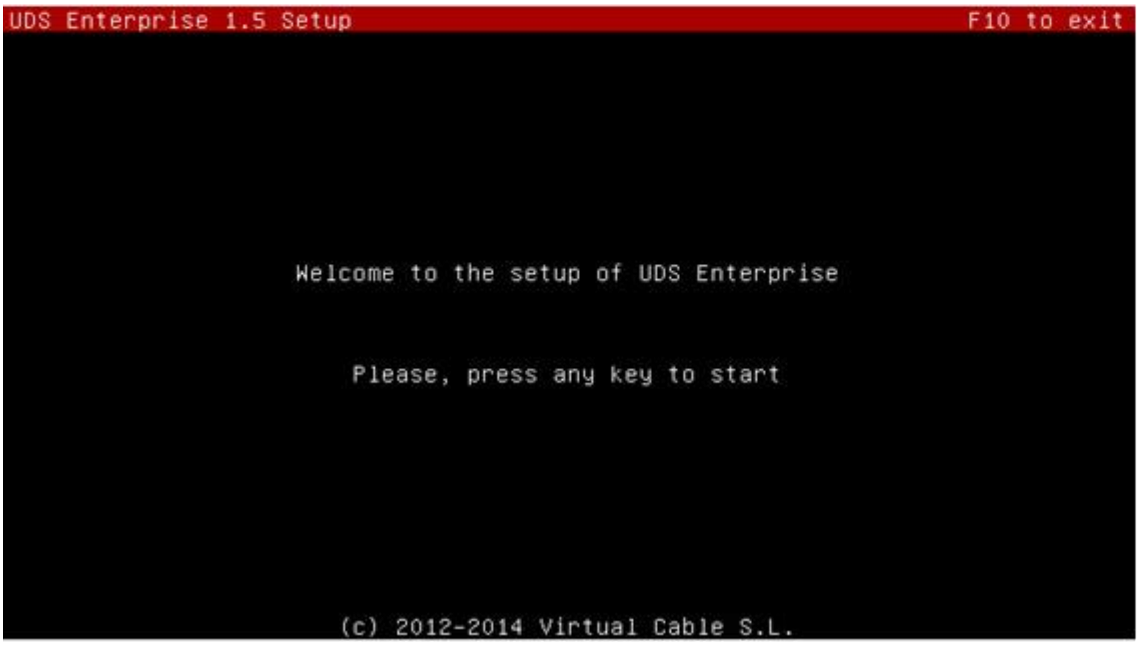
3.2 UDS Platform Installation

3.2.1 UDS Broker Installation

Once the virtual machine that makes up the broker is turned on, a client console will be opened to access the virtual machine.

NOTE: In order to successfully configure a UDS Broker server, a MySQL database server with a completely empty database must be configured.

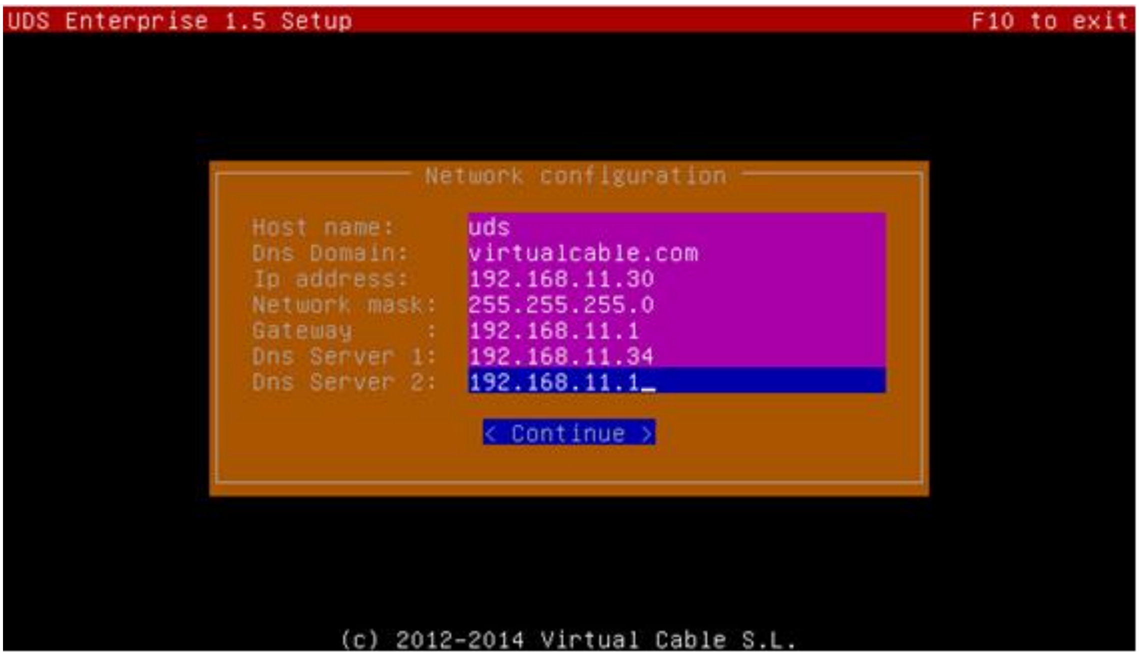
At this moment the UDS broker server configuration process starts:



```
UDS Enterprise 1.5 Setup                                          F10 to exit


                Welcome to the setup of UDS Enterprise

                    Please, press any key to start


                   (c) 2012-2014 Virtual Cable S.L.
```

Step 1.- The following parameters will be configured:

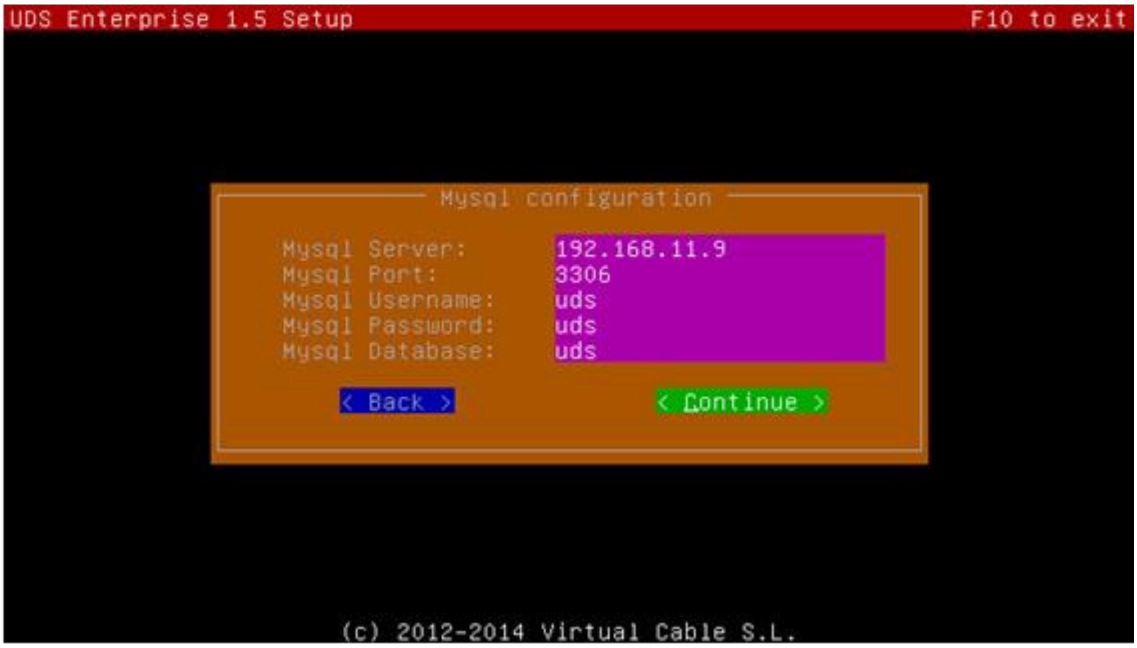
- **DNS name.** This name must be defined in the corresponding DNS server.
- **Domain.** Domain where the UDS Broker server will be hosted.
- **UDS Broker server network data.** Fill in the following data:
    - **Server IP**
    - **Network mask**
    - **Gateway IP**
    - **DNS servers IPs**



```
UDS Enterprise 1.5 Setup                                          F10 to exit

            Network configuration
  Host name:       uds
  Dns Domain:      virtualcable.com
  Ip address:      192.168.11.30
  Network mask:    255.255.255.0
  Gateway     :    192.168.11.1
  Dns Server 1:    192.168.11.34
  Dns Server 2:    192.168.11.1_

                  < Continue >

                     (c) 2012-2014 Virtual Cable S.L.
```

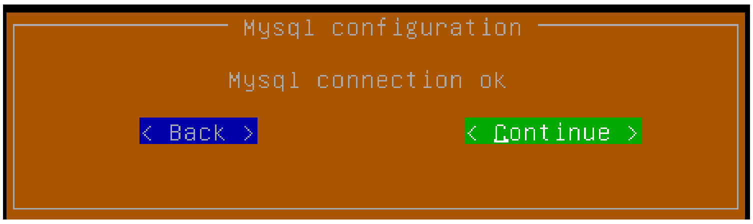
Step 2.- Fill in the database server connection data.

- **MySQL server:** IP address or database server DNS name.
- **MySQL port:** Connection port to the MySQL server. By default: 3306.
- **MySQL user:** Database administrator user.
- **MySQL Password:** User password previously defined.
- **MySQL database:** Database where the table structure will be created to host the data needed for the UDS platform.



```
UDS Enterprise 1.5 Setup                                    F10 to exit

                    Mysql configuration
    Mysql Server:      192.168.11.9
    Mysql Port:        3306
    Mysql Username:    uds
    Mysql Password:    uds
    Mysql Database:    uds

              (c) 2012-2014 Virtual Cable S.L.
```

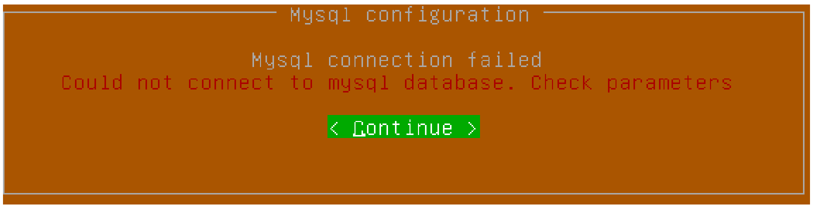
Once the required information has been filled out, click on **<Continue>** and the system will perform a connection test to the database with the information you have filled in.



Mysql configuration

Mysql connection ok

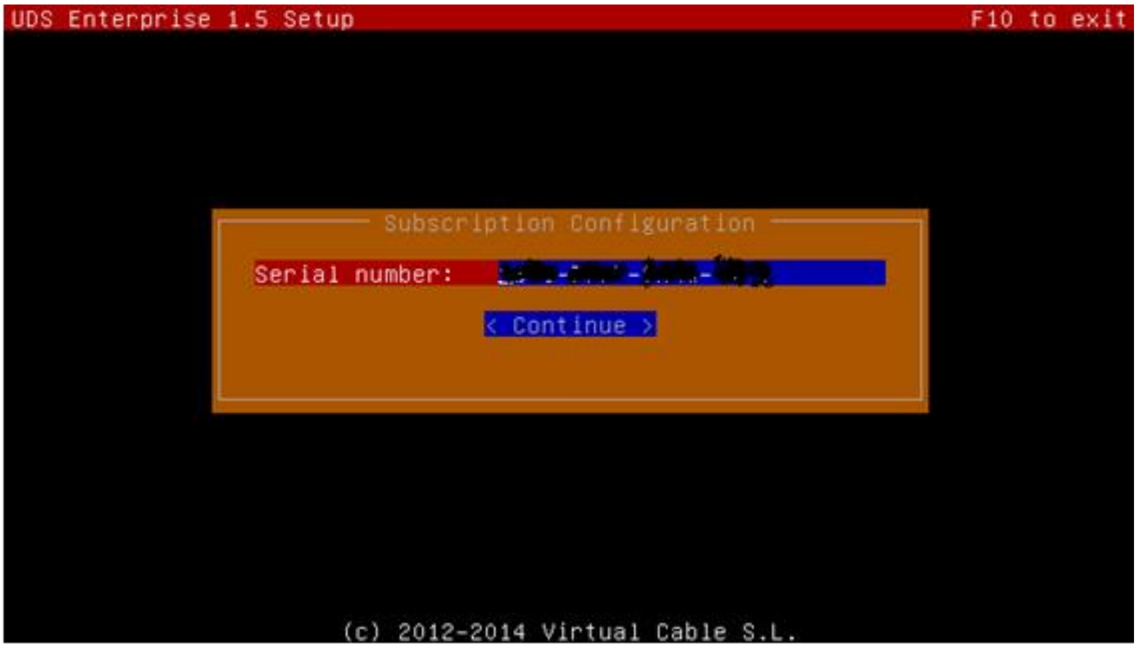
In the event that this fails, a connection error screen will appear.



Mysql configuration

Mysql connection failed
Could not connect to mysql database. Check parameters

Step 3.- Enter the subscription activation code.



UDS Enterprise 1.5 Setup F10 to exit

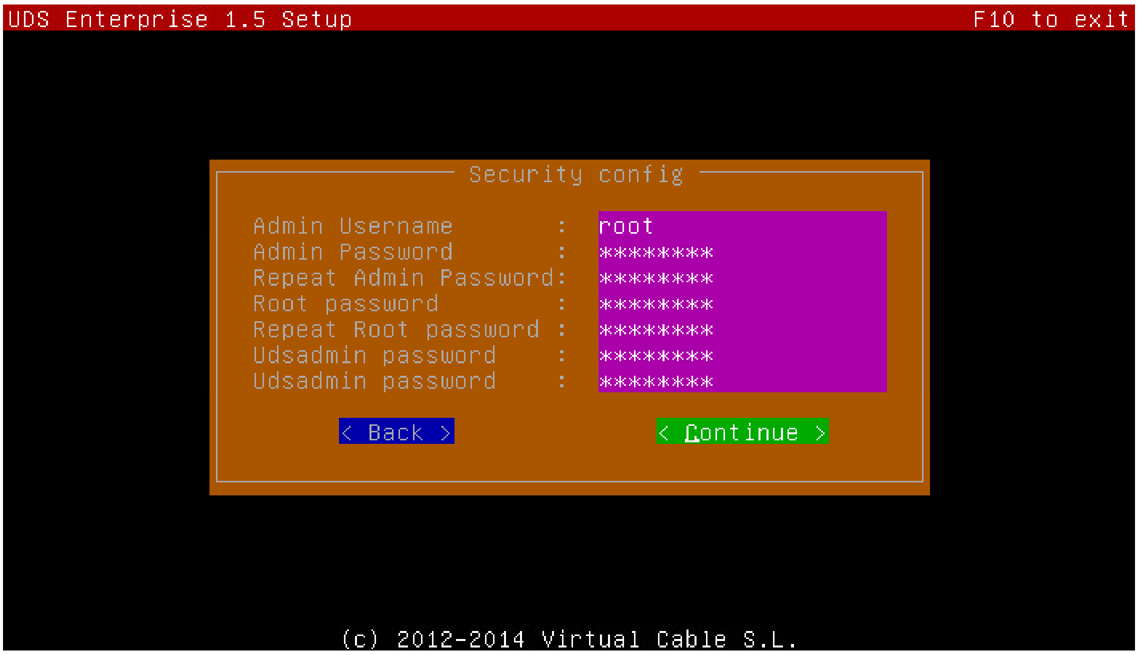
Subscription Configuration

Serial number: [redacted]

(c) 2012-2014 Virtual Cable S.L.

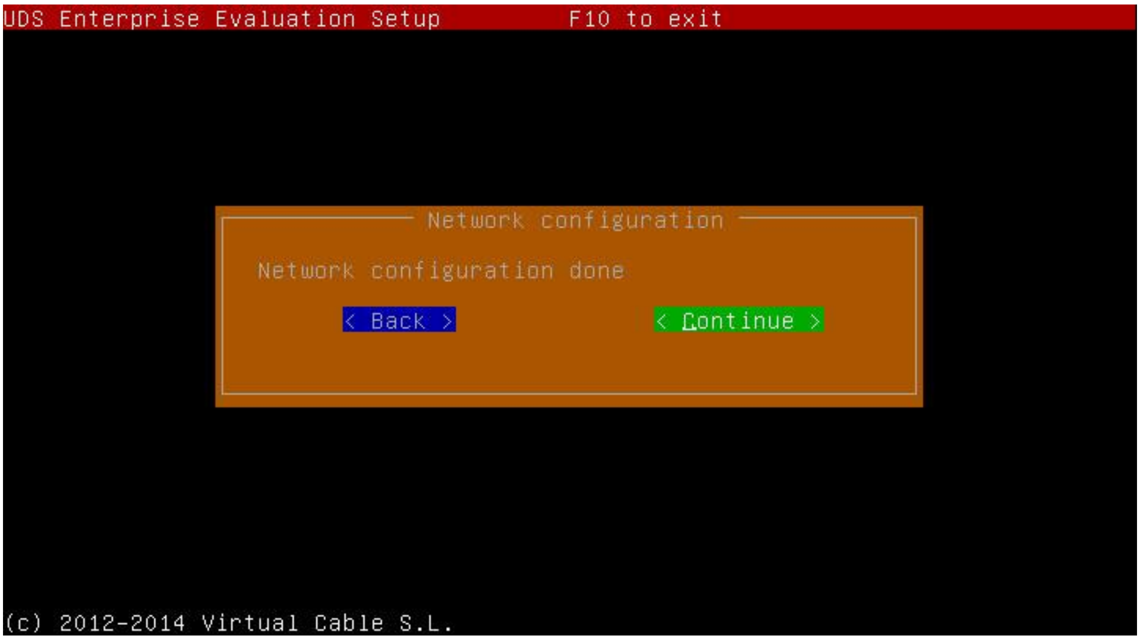
Step 4.- Configuration of UDS server user accesses and passwords. In this step, a username will be created in order to access the UDS Administration. The information to fill in is the following:

- **User Admin:** User with administrator permission for managing the UDS platform.
- **Password Admin:** Password of the administrating user created in the previous step.
- **Root Password:** Password of the root user in order to access the UDS Broker server from the VMwre client console.
- **Udsadmin Password:** Password of the **Udsadmin** user that will allow you to make an SSH connection to the UDS server machine.



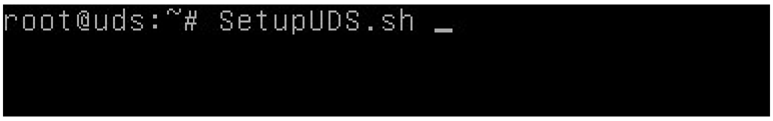
```
UDS Enterprise 1.5 Setup                                         F10 to exit

                          Security config

          Admin Username         :  root
          Admin Password         :  *********
          Repeat Admin Password:    *********
          Root password          :  *********
          Repeat Root password   :  *********
          Udsadmin password      :  *********
          Udsadmin password      :  *********

                < Back >                  < Continue >

                     (c) 2012-2014 Virtual Cable S.L.
```

Once all of these steps have been completed, you will need to restart the UDS Broker server.



```
UDS Enterprise Evaluation Setup          F10 to exit

                 Network configuration
    Network configuration done
         < Back >                 < Continue >

(c) 2012-2014 Virtual Cable S.L.
```

If a rerun of the configuration wizard is needed, we will have to validate us in the server using the credentials by default and we run “SetupUDS.sh”.



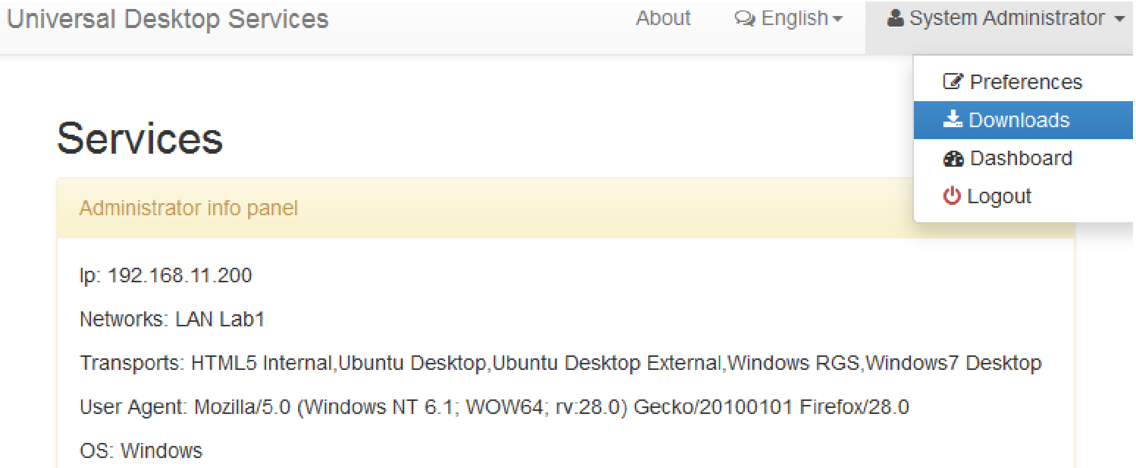
```
root@uds:~# SetupUDS.sh _
```

Once the new data are entered, we will have to manually restart the server using “Reboot” command.

3.2.2 UDS Administration client installation

UDS supplies an administration client to perform all system configurations. From version 1.5 onwards, the platform administration can be managed using web administration, and we strongly recommend this new way of administration. In future versions the only way to administer UDS will be through web.

To download the thick administration client, you must have the Broker UDS server configured (See section 3.2.1) and we access through web. We enter the system with the “root” user or with other user with administration rights on the system and in the user menu we choose “Downloads”.



Ip: 192.168.11.200

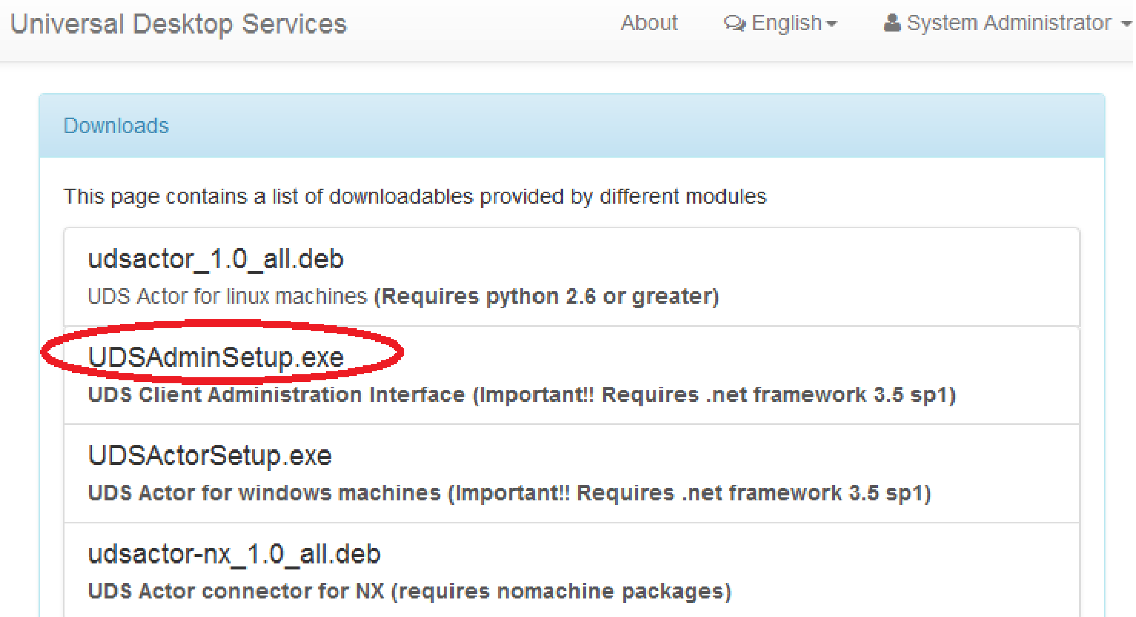
Networks: LAN Lab1

Transports: HTML5 Internal,Ubuntu Desktop,Ubuntu Desktop External,Windows RGS,Windows7 Desktop

User Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:28.0) Gecko/20100101 Firefox/28.0

OS: Windows

We select and download "UDSAdminSetup.exe".



Universal Desktop Services

Downloads

This page contains a list of downloadables provided by different modules

udsactor_1.0_all.deb

UDS Actor for linux machines **(Requires python 2.6 or greater)**

UDSAdminSetup.exe

UDS Client Administration Interface (Important!! Requires .net framework 3.5 sp1)

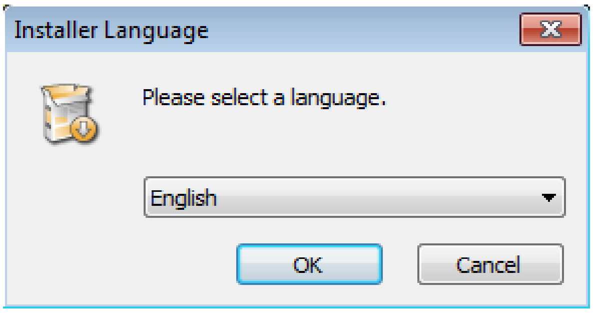
UDSActorSetup.exe

UDS Actor for windows machines (Important!! Requires .net framework 3.5 sp1)

udsactor-nx_1.0_all.deb

UDS Actor connector for NX (requires nomachine packages)

Once downloaded, you can run the installation program and select the installation language.

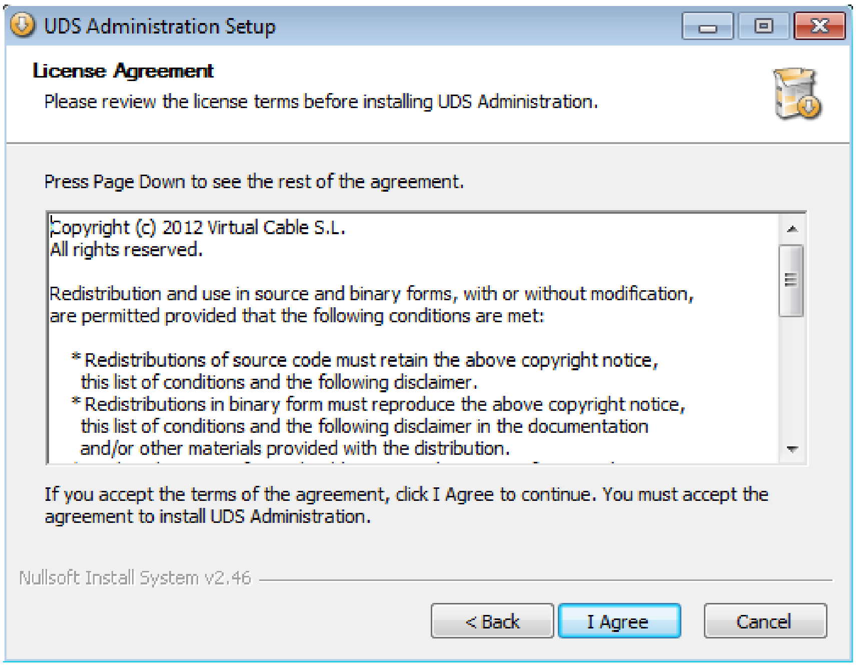


Installer Language

Please select a language.

English

In the next dialogue window we accept the license agreement:



UDS Administration Setup

License Agreement

Please review the license terms before installing UDS Administration.

Press Page Down to see the rest of the agreement.

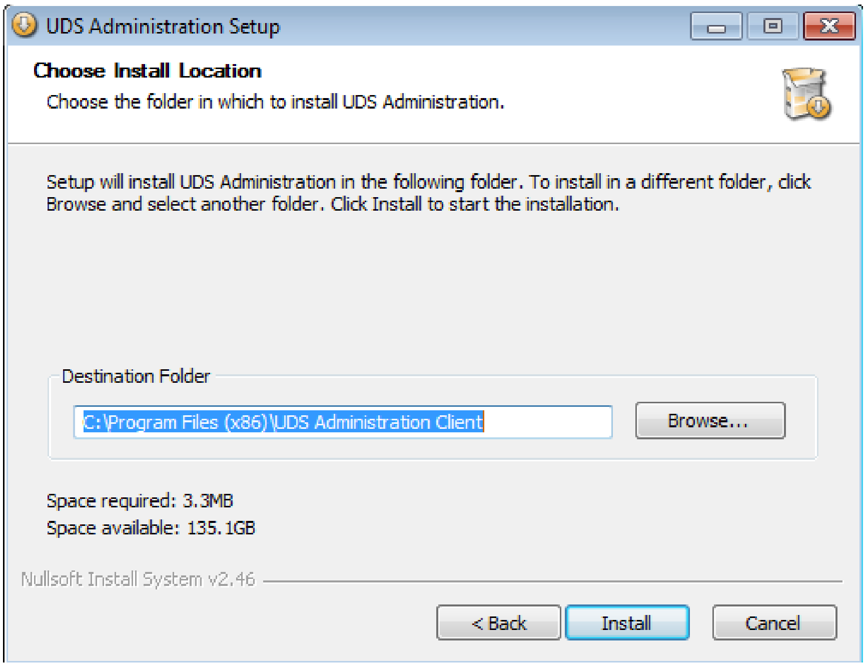
Copyright (c) 2012 Virtual Cable S.L.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install UDS Administration.

We select the installation location of the administration client files:



UDS Administration Setup

Choose Install Location

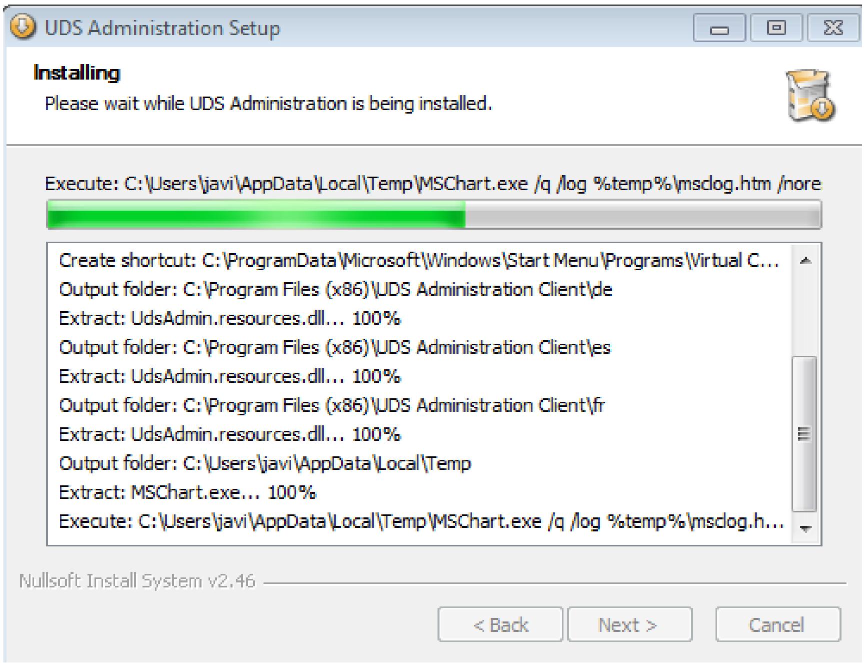
Choose the folder in which to install UDS Administration.

Setup will install UDS Administration in the following folder. To install in a different folder, click Browse and select another folder. Click Install to start the installation.

Destination Folder: C:\Program Files (x86)\UDS Administration Client

Space required: 3.3MB
Space available: 135.1GB

The next dialogue window shows the installation status until it is finished:



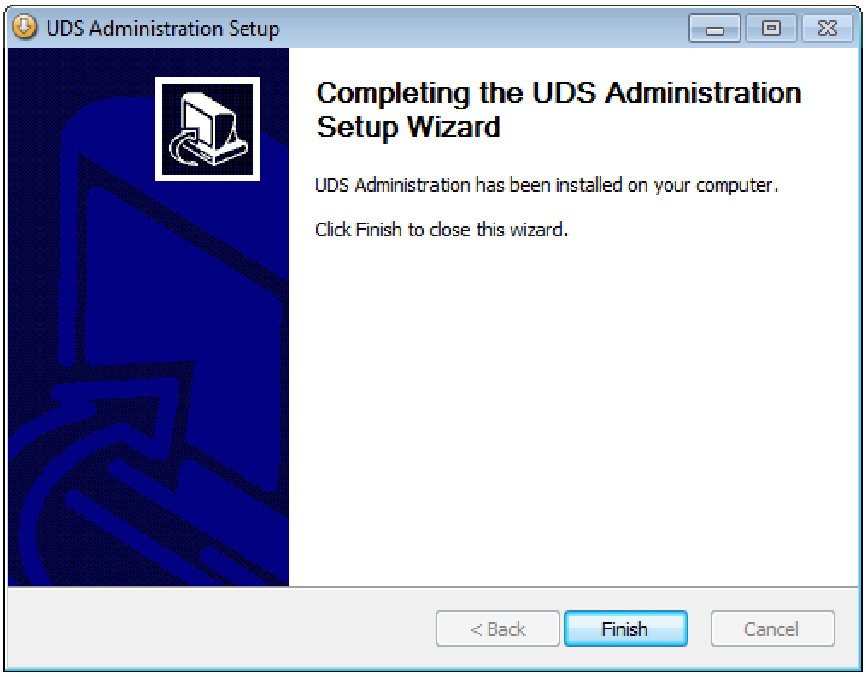
Installing

Please wait while UDS Administration is being installed.

```
Execute: C:\Users\javi\AppData\Local\Temp\MSChart.exe /q /log %temp%\msclog.htm /nore
```

```
Create shortcut: C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Virtual C...
Output folder: C:\Program Files (x86)\UDS Administration Client\de
Extract: UdsAdmin.resources.dll... 100%
Output folder: C:\Program Files (x86)\UDS Administration Client\es
Extract: UdsAdmin.resources.dll... 100%
Output folder: C:\Program Files (x86)\UDS Administration Client\fr
Extract: UdsAdmin.resources.dll... 100%
Output folder: C:\Users\javi\AppData\Local\Temp
Extract: MSChart.exe... 100%
Execute: C:\Users\javi\AppData\Local\Temp\MSChart.exe /q /log %temp%\msclog.h...
```

If there were no installation errors, the UDS administration client installation process is wrapped up:



Completing the UDS Administration Setup Wizard

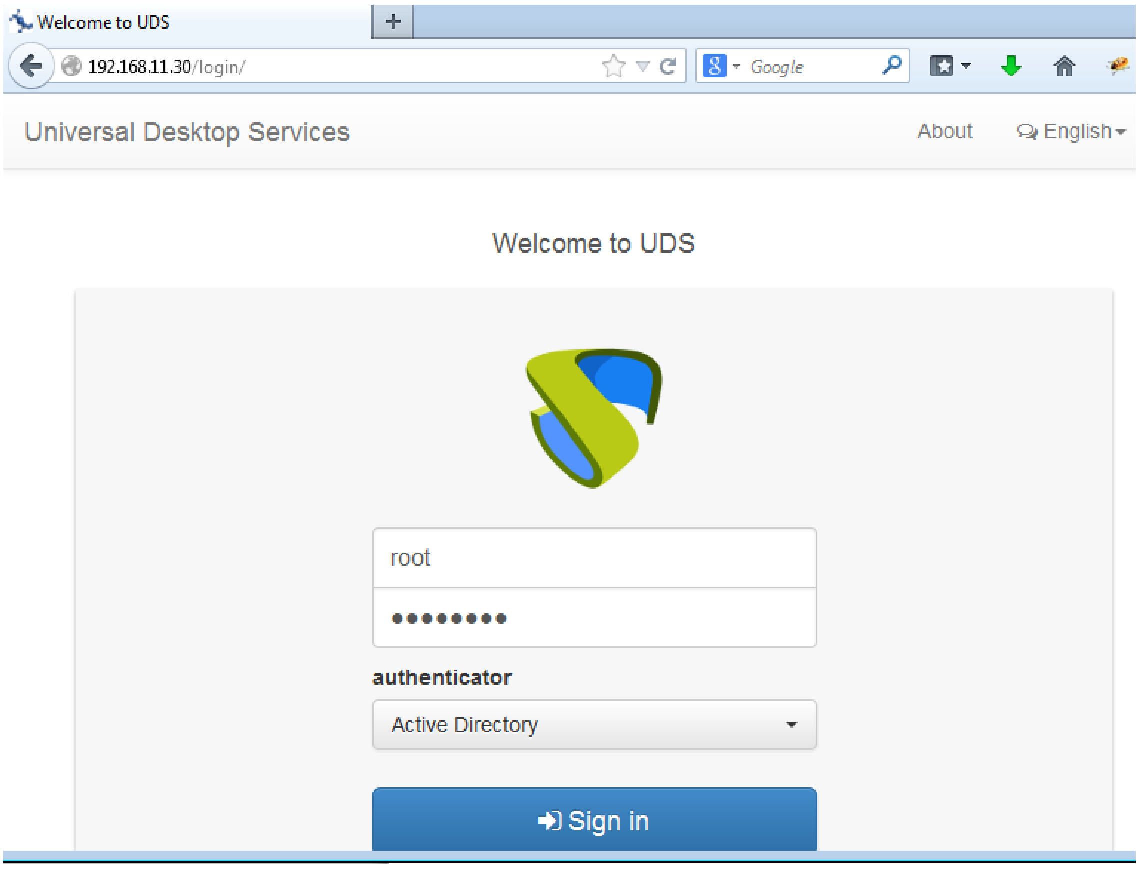
UDS Administration has been installed on your computer.

Click Finish to close this wizard.

3.2.3 UDS actor installation and configuration

In order to install the UDS actor, you must have previously downloaded the UDS Broker by selecting the suitable actor for each platform (Windows and Linux).

In order to do that, you connect to the UDS Broker via web browser and using root user credentials (if active) or a user with **administration privileges** to access the downloads.



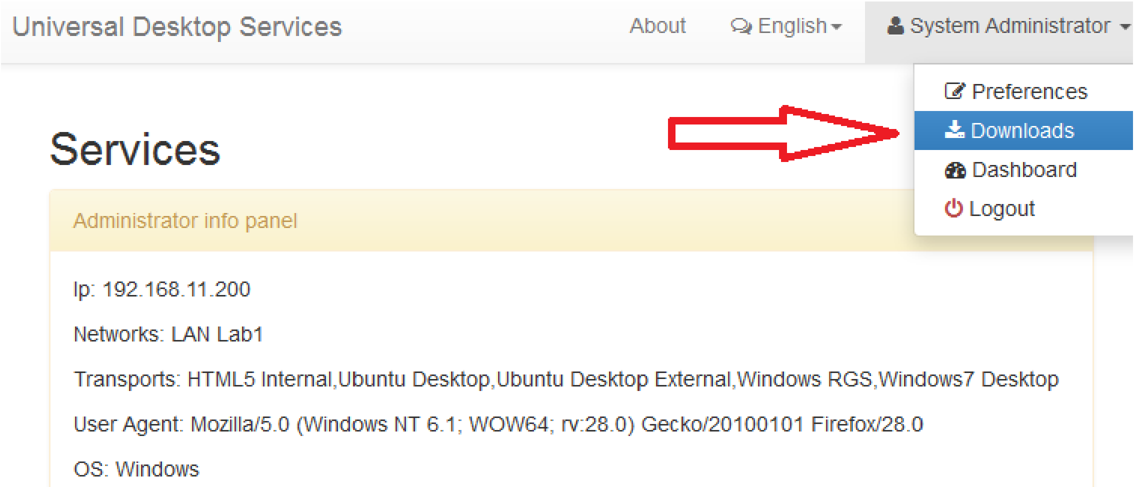
root

●●●●●●●●

authenticator

Active Directory

In the user menu we choose "Downloads":



Services

Administrator info panel

Ip: 192.168.11.200

Networks: LAN Lab1

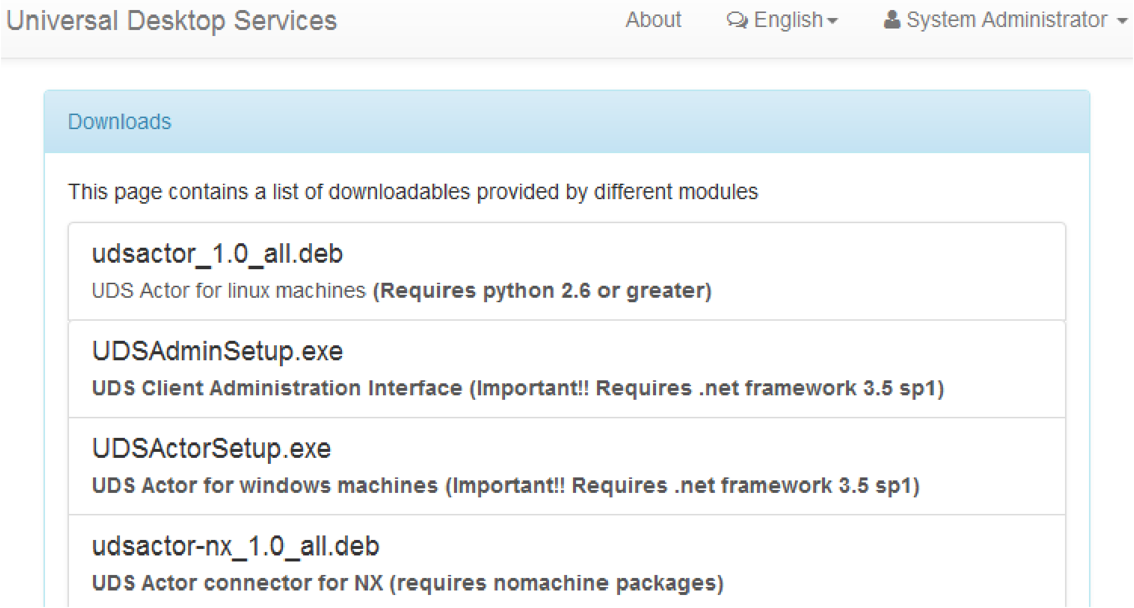
Transports: HTML5 Internal,Ubuntu Desktop,Ubuntu Desktop External,Windows RGS,Windows7 Desktop

User Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:28.0) Gecko/20100101 Firefox/28.0

OS: Windows

The UDS actors available for download will be shown in the browser. Select the actor corresponding to the operating system that is installed on the template on which the desktops are going to be deployed:

- **Udsactor_1.0_all.deb :** Actor for Linux machines.
- **UDSActorSetup.exe:** Actor for Windows machines.
- **Udsactor-nx_1.0_all.deb:** UDS connector for NX in Linux machines.



Downloads

This page contains a list of downloadables provided by different modules

udsactor_1.0_all.deb
UDS Actor for linux machines **(Requires python 2.6 or greater)**

UDSAdminSetup.exe
UDS Client Administration Interface (Important!! Requires .net framework 3.5 sp1)

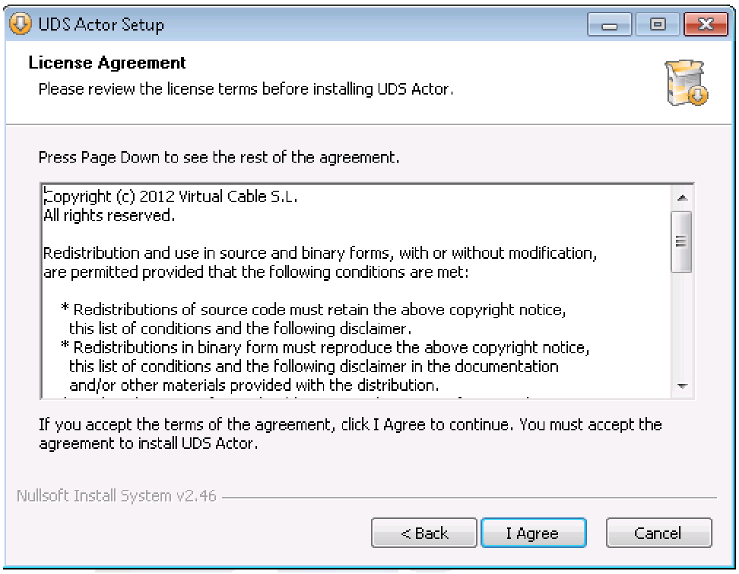
UDSActorSetup.exe
UDS Actor for windows machines (Important!! Requires .net framework 3.5 sp1)

udsactor-nx_1.0_all.deb
UDS Actor connector for NX (requires nomachine packages)

3.2.3.1 UDS Windows Actor Installation

Once the UDS actor installation program has been transferred to the template, it can now be installed:

In the first dialogue window, we accept the license agreement:



UDS Actor Setup

License Agreement

Please review the license terms before installing UDS Actor.

Press Page Down to see the rest of the agreement.

Copyright (c) 2012 Virtual Cable S.L.
All rights reserved.

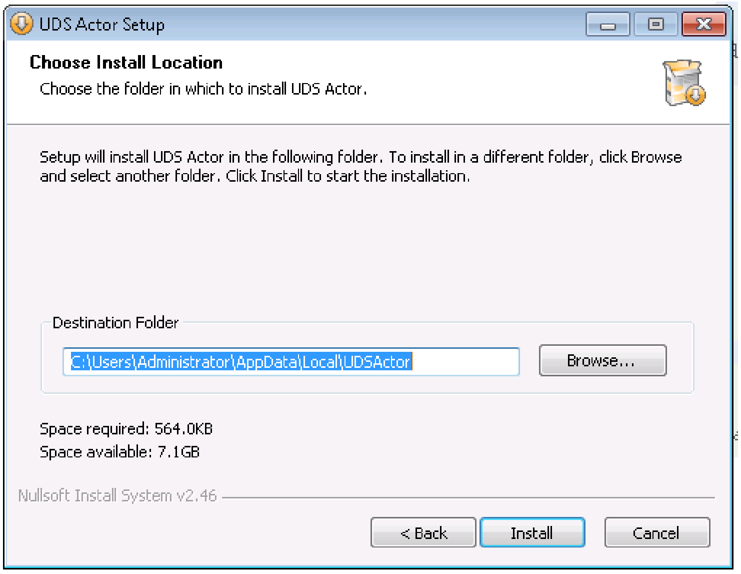
Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

 * Redistributions of source code must retain the above copyright notice,
 this list of conditions and the following disclaimer.
 * Redistributions in binary form must reproduce the above copyright notice,
 this list of conditions and the following disclaimer in the documentation
 and/or other materials provided with the distribution.

If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install UDS Actor.

Nullsoft Install System v2.46

In the next dialogue window, we select the installation location for the UDS actor:



UDS Actor Setup

Choose Install Location

Choose the folder in which to install UDS Actor.

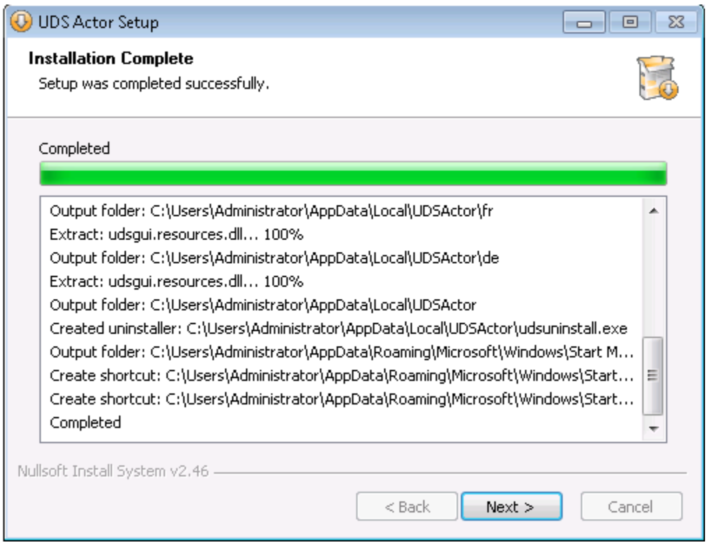
Setup will install UDS Actor in the following folder. To install in a different folder, click Browse and select another folder. Click Install to start the installation.

Destination Folder: C:\Users\Administrator\AppData\Local\UDSActor

Space required: 564.0KB
Space available: 7.1GB

Nullsoft Install System v2.46

Click on **"Install"** and the UDS actor will begin its installation process:



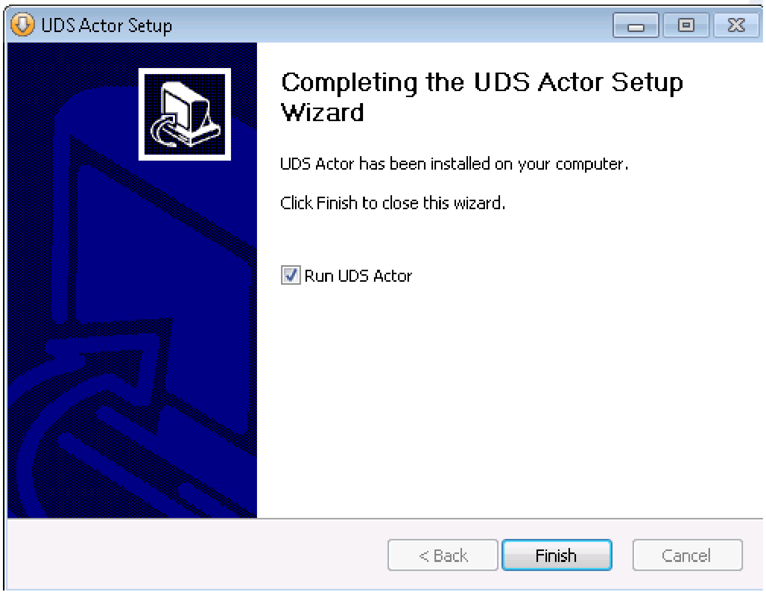
Installation Complete

Setup was completed successfully.

Completed

```
Output folder: C:\Users\Administrator\AppData\Local\UDSActor\fr
Extract: udsgui.resources.dll... 100%
Output folder: C:\Users\Administrator\AppData\Local\UDSActor\de
Extract: udsgui.resources.dll... 100%
Output folder: C:\Users\Administrator\AppData\Local\UDSActor
Created uninstaller: C:\Users\Administrator\AppData\Local\UDSActor\udsuninstall.exe
Output folder: C:\Users\Administrator\AppData\Roaming\Microsoft\Windows\Start M...
Create shortcut: C:\Users\Administrator\AppData\Roaming\Microsoft\Windows\Start...
Create shortcut: C:\Users\Administrator\AppData\Roaming\Microsoft\Windows\Start...
Completed
```

Once the installation has been completed, the UDS actor is configured:



Completing the UDS Actor Setup Wizard

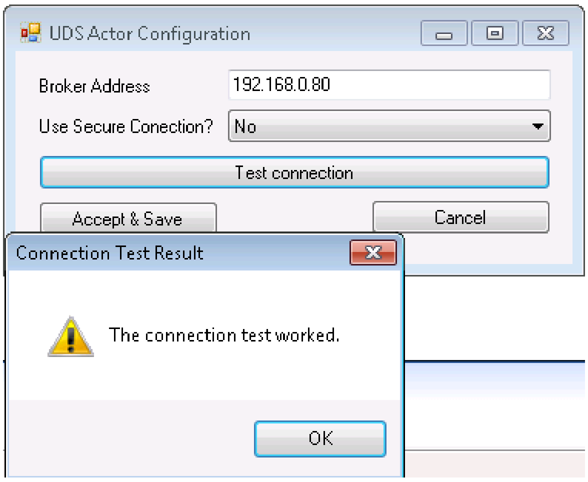
UDS Actor has been installed on your computer.

Click Finish to close this wizard.

☑ Run UDS Actor

Enter the IP address of the UDS Broker and indicate whether or not you will use a secure connection.

Once these parameters have been configured, run the connection test to check the connectivity with the broker server.



Broker Address: 192.168.0.80

Use Secure Conection?: No

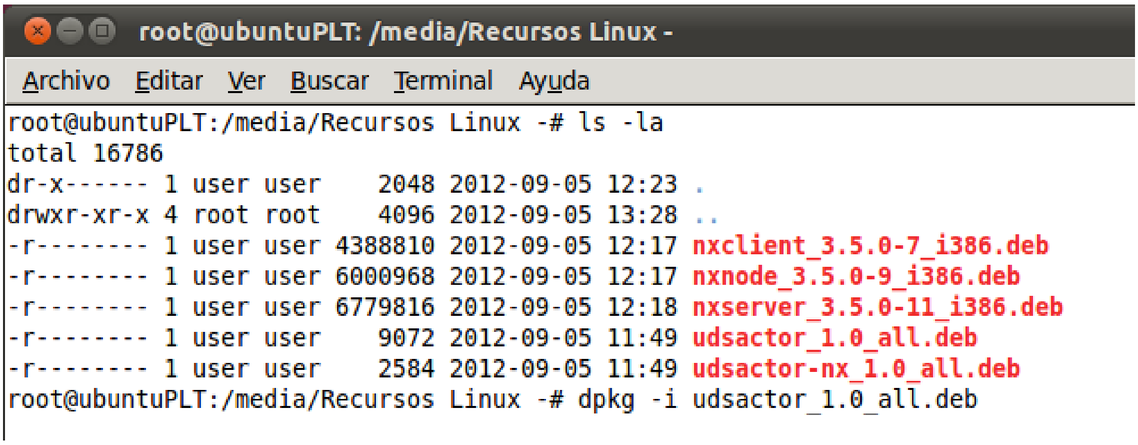
The connection test worked.

It will be necessary to set up the remote access to the desktop in order to make the connection through RDP (recommended transport for Windows machines).

Once the UDS actor has been configured, the virtual machine is now ready to be used as an UDS system virtual desktop template.

3.2.3.2 Linux UDS Actor Installation

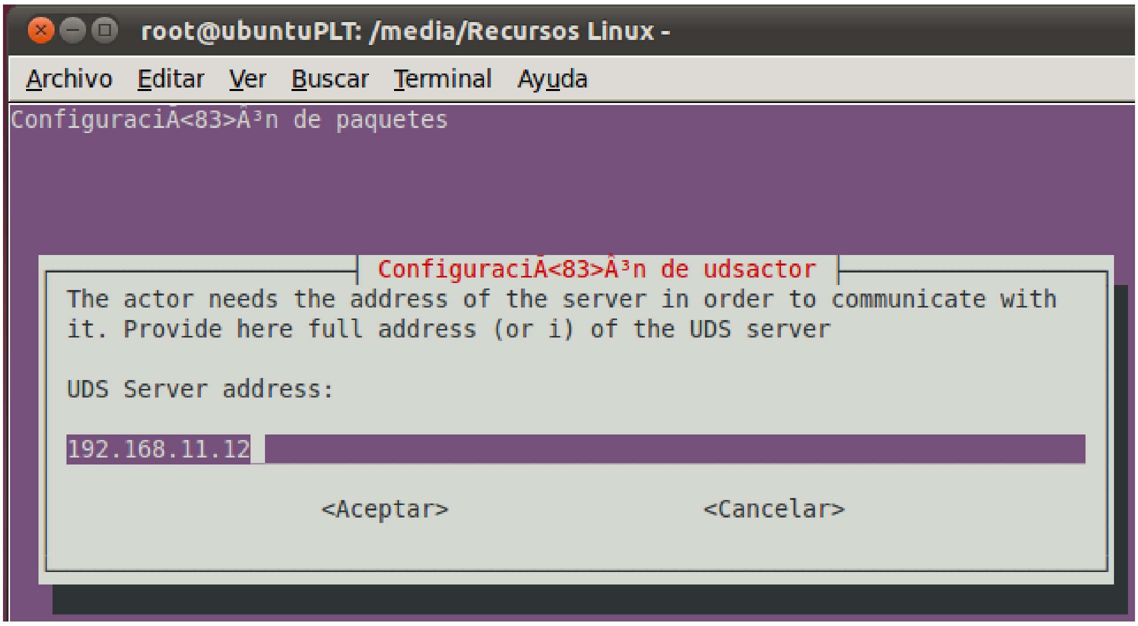
Once the UDS actor installation suite has been transferred to the template, the installation begins by running the file udsactor_1.0_all.deb



root@ubuntuPLT: /media/Recursos Linux -
Archivo Editar Ver Buscar Terminal Ayuda

```
root@ubuntuPLT:/media/Recursos Linux -# ls -la
total 16786
dr-x------ 1 user user    2048 2012-09-05 12:23 .
drwxr-xr-x 4 root root    4096 2012-09-05 13:28 ..
-r-------- 1 user user 4388810 2012-09-05 12:17 nxclient_3.5.0-7_i386.deb
-r-------- 1 user user 6000968 2012-09-05 12:17 nxnode_3.5.0-9_i386.deb
-r-------- 1 user user 6779816 2012-09-05 12:18 nxserver_3.5.0-11_i386.deb
-r-------- 1 user user    9072 2012-09-05 11:49 udsactor_1.0_all.deb
-r-------- 1 user user    2584 2012-09-05 11:49 udsactor-nx_1.0_all.deb
root@ubuntuPLT:/media/Recursos Linux -# dpkg -i udsactor_1.0_all.deb
```

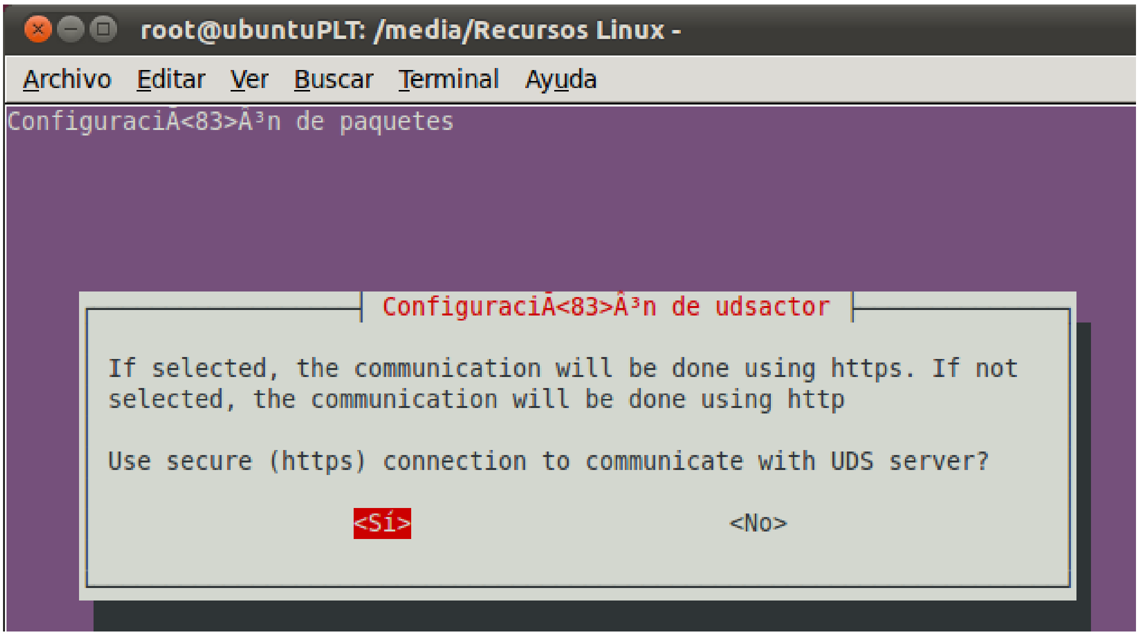
Enter the UDS server address:



root@ubuntuPLT: /media/Recursos Linux -
Archivo Editar Ver Buscar Terminal Ayuda

```
ConfiguraciÃ<83>Â³n de paquetes

            ConfiguraciÃ<83>Â³n de udsactor
 The actor needs the address of the server in order to communicate with
 it. Provide here full address (or i) of the UDS server

 UDS Server address:

 192.168.11.12

              <Aceptar>                    <Cancelar>
```

Select how communication with the UDS broker will be carried out.


root@ubuntuPLT: /media/Recursos Linux -
Archivo Editar Ver Buscar Terminal Ayuda

```
ConfiguraciÃ<83>Â³n de paquetes

ConfiguraciÃ<83>Â³n de udsactor

If selected, the communication will be done using https. If not
selected, the communication will be done using http

Use secure (https) connection to communicate with UDS server?

<Sí>                    <No>
```

Once the installation of the UDS actor has been completed, the installation of NX 3.5 software is recommended. This software is available in the following link:

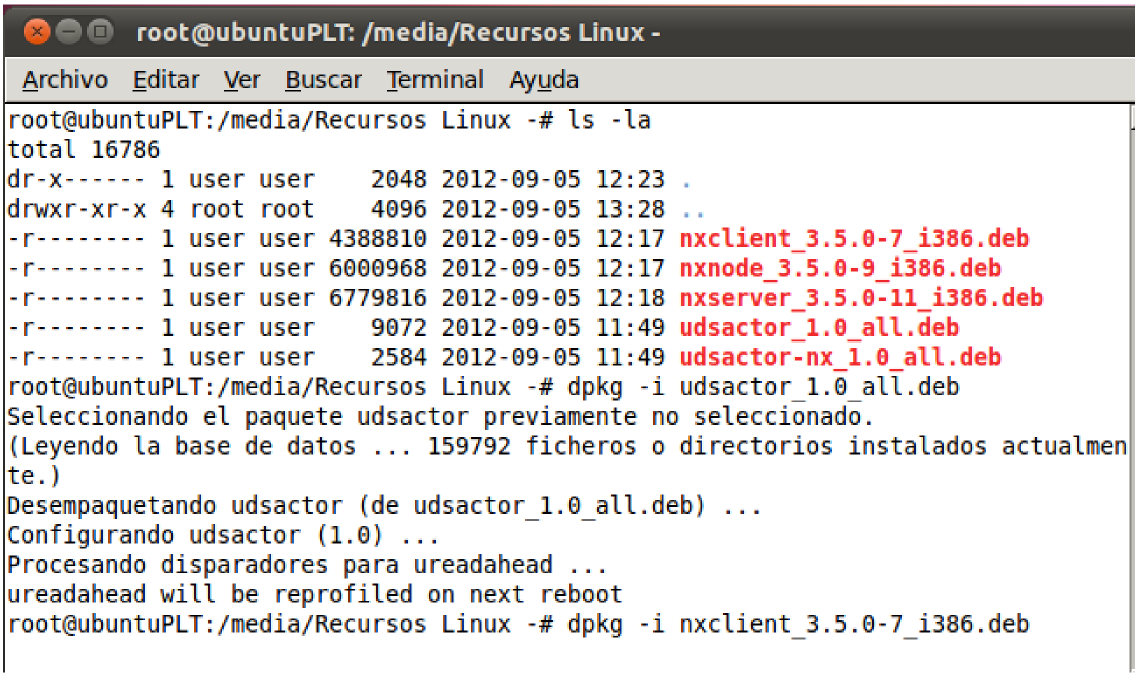
http://www.nomachine.com/select-package.php?os=linux&id=1

The installation order of the different packets is:

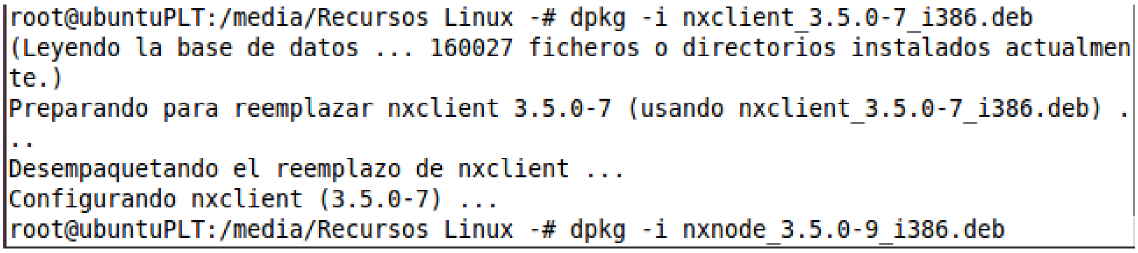
1° “nxclient_x” packet

2° “nxnode_x” packet

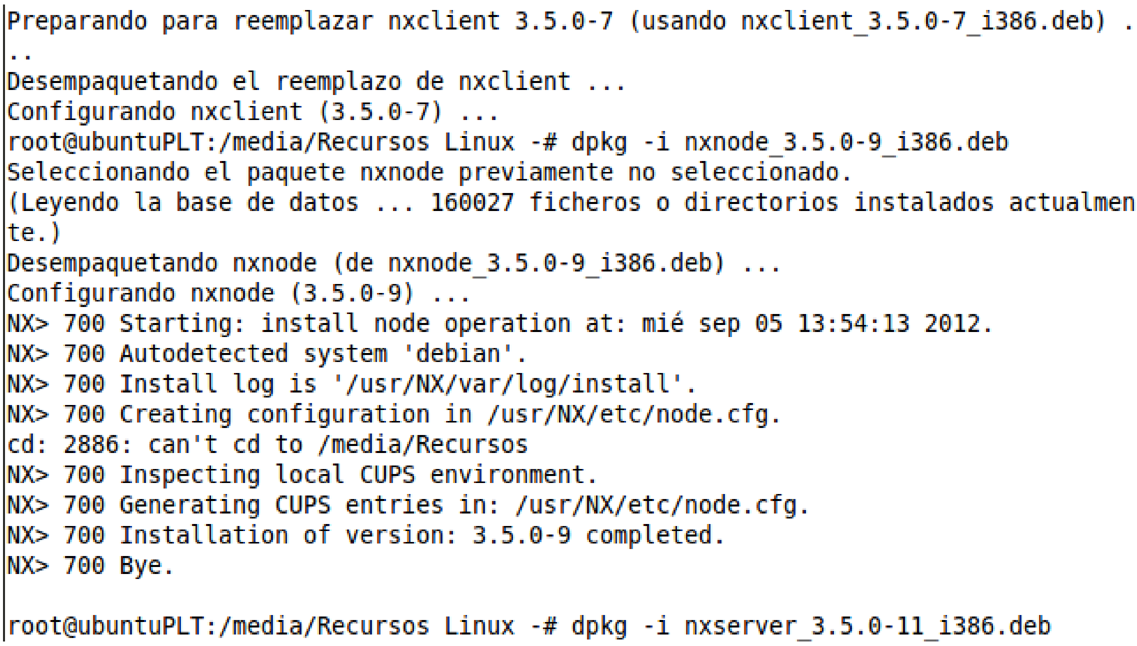
3° “nxserver_x” packet


root@ubuntuPLT: /media/Recursos Linux -
Archivo Editar Ver Buscar Terminal Ayuda

```
root@ubuntuPLT:/media/Recursos Linux -# ls -la
total 16786
dr-x------ 1 user user    2048 2012-09-05 12:23 .
drwxr-xr-x 4 root root    4096 2012-09-05 13:28 ..
-r-------- 1 user user 4388810 2012-09-05 12:17 nxclient_3.5.0-7_i386.deb
-r-------- 1 user user 6000968 2012-09-05 12:17 nxnode_3.5.0-9_i386.deb
-r-------- 1 user user 6779816 2012-09-05 12:18 nxserver_3.5.0-11_i386.deb
-r-------- 1 user user    9072 2012-09-05 11:49 udsactor_1.0_all.deb
-r-------- 1 user user    2584 2012-09-05 11:49 udsactor-nx_1.0_all.deb
root@ubuntuPLT:/media/Recursos Linux -# dpkg -i udsactor_1.0_all.deb
Seleccionando el paquete udsactor previamente no seleccionado.
(Leyendo la base de datos ... 159792 ficheros o directorios instalados actualmen
te.)
Desempaquetando udsactor (de udsactor_1.0_all.deb) ...
Configurando udsactor (1.0) ...
Procesando disparadores para ureadahead ...
ureadahead will be reprofiled on next reboot
root@ubuntuPLT:/media/Recursos Linux -# dpkg -i nxclient_3.5.0-7_i386.deb
```

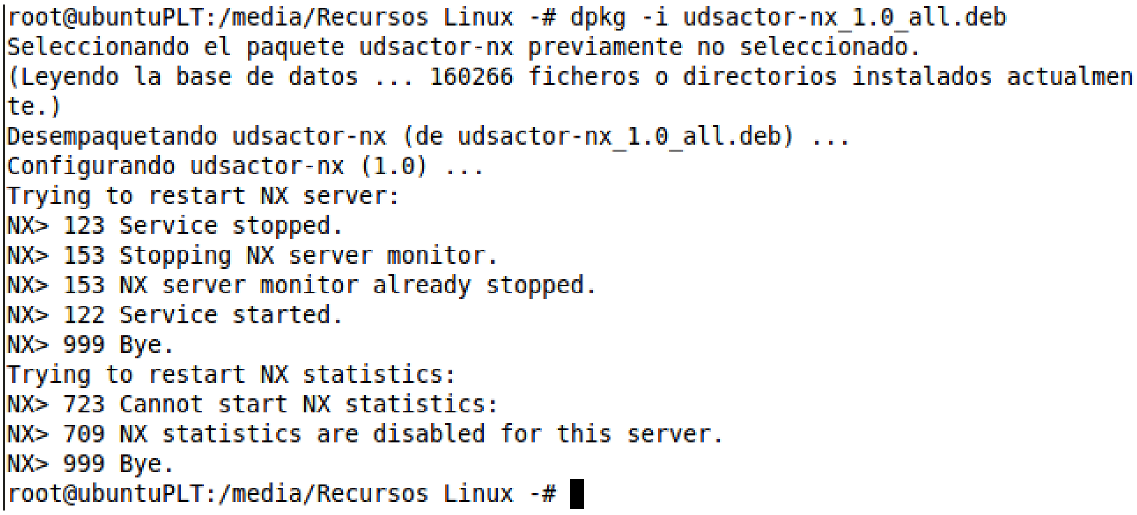


```
root@ubuntuPLT:/media/Recursos Linux -# dpkg -i nxclient_3.5.0-7_i386.deb
(Leyendo la base de datos ... 160027 ficheros o directorios instalados actualmen
te.)
Preparando para reemplazar nxclient 3.5.0-7 (usando nxclient_3.5.0-7_i386.deb) .
..
Desempaquetando el reemplazo de nxclient ...
Configurando nxclient (3.5.0-7) ...
root@ubuntuPLT:/media/Recursos Linux -# dpkg -i nxnode_3.5.0-9_i386.deb
```



```
Preparando para reemplazar nxclient 3.5.0-7 (usando nxclient_3.5.0-7_i386.deb) .
..
Desempaquetando el reemplazo de nxclient ...
Configurando nxclient (3.5.0-7) ...
root@ubuntuPLT:/media/Recursos Linux -# dpkg -i nxnode_3.5.0-9_i386.deb
Seleccionando el paquete nxnode previamente no seleccionado.
(Leyendo la base de datos ... 160027 ficheros o directorios instalados actualmen
te.)
Desempaquetando nxnode (de nxnode_3.5.0-9_i386.deb) ...
Configurando nxnode (3.5.0-9) ...
NX> 700 Starting: install node operation at: mié sep 05 13:54:13 2012.
NX> 700 Autodetected system 'debian'.
NX> 700 Install log is '/usr/NX/var/log/install'.
NX> 700 Creating configuration in /usr/NX/etc/node.cfg.
cd: 2886: can't cd to /media/Recursos
NX> 700 Inspecting local CUPS environment.
NX> 700 Generating CUPS entries in: /usr/NX/etc/node.cfg.
NX> 700 Installation of version: 3.5.0-9 completed.
NX> 700 Bye.

root@ubuntuPLT:/media/Recursos Linux -# dpkg -i nxserver_3.5.0-11_i386.deb
```

It is also necessary to install the "udsactor-nx_1.0_all.deb" packet so that the system can identify the active connections of the NX connection.



```
root@ubuntuPLT:/media/Recursos Linux -# dpkg -i udsactor-nx_1.0_all.deb
Seleccionando el paquete udsactor-nx previamente no seleccionado.
(Leyendo la base de datos ... 160266 ficheros o directorios instalados actualmen
te.)
Desempaquetando udsactor-nx (de udsactor-nx_1.0_all.deb) ...
Configurando udsactor-nx (1.0) ...
Trying to restart NX server:
NX> 123 Service stopped.
NX> 153 Stopping NX server monitor.
NX> 153 NX server monitor already stopped.
NX> 122 Service started.
NX> 999 Bye.
Trying to restart NX statistics:
NX> 723 Cannot start NX statistics:
NX> 709 NX statistics are disabled for this server.
NX> 999 Bye.
root@ubuntuPLT:/media/Recursos Linux -# █
```

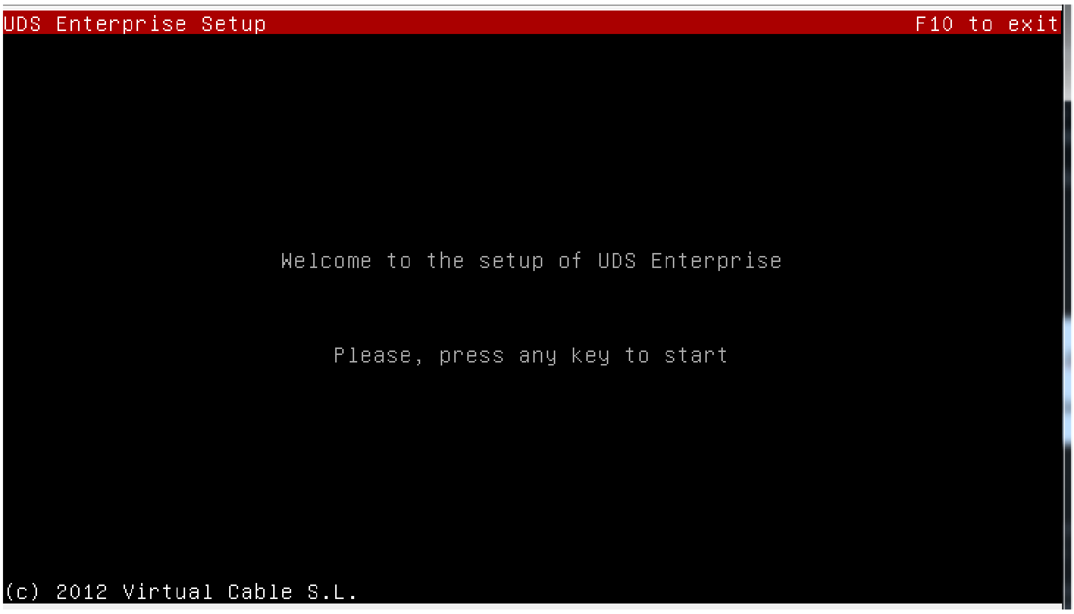
Once the UDS actor has been configured, the virtual machine is now ready to be used as a UDS system virtual desktop template.

3.2.4 UDS Tunneler Installation

Once the virtual machine that makes up the UDS tunneler is turned on, a client console will be opened to access the virtual machine.

NOTE: To configure UDS Tunneler successfully you must configure UDS Broker beforehand.

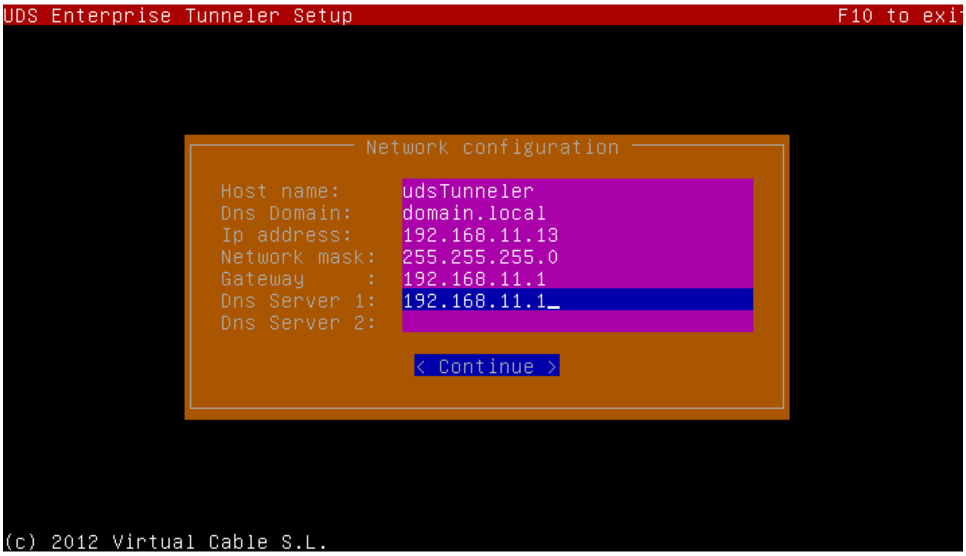
At this moment the UDS tunneler configuration process starts:



```
UDS Enterprise Setup                                                F10 to exit




                    Welcome to the setup of UDS Enterprise


                        Please, press any key to start





(c) 2012 Virtual Cable S.L.
```

Step 1.- The following parameters will be configured:

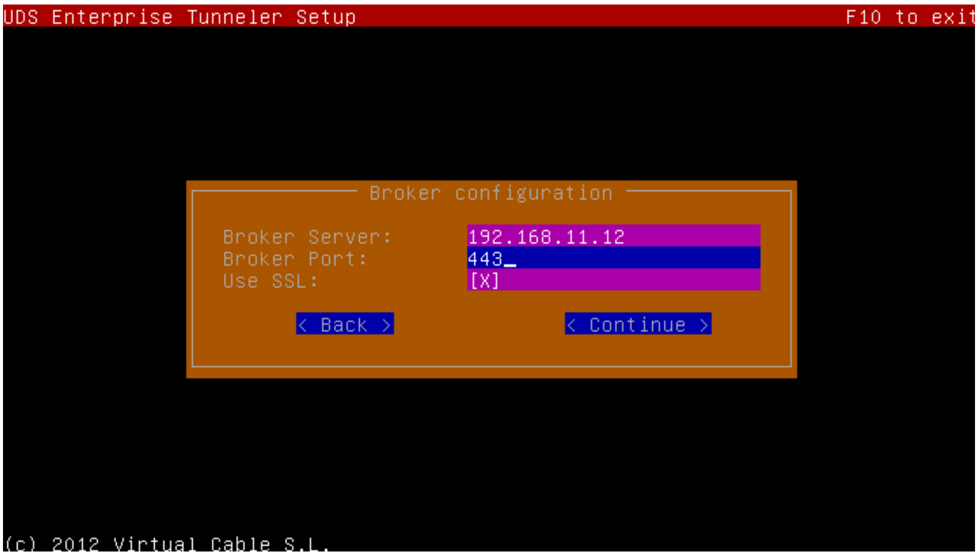
- **DNS name.** This name must be defined in the corresponding DNS server.
- **Dominio**. Domain where the UDS Tunneler server will be hosted.
- UDS Tunneler server **network data**. Fill in the following data:
  - **Server IP**
  - **Network mask**
  - **Gateway IP**
  - **DNS servers IPs**



```
UDS Enterprise Tunneler Setup                                F10 to exi

                    Network configuration

     Host name:       udsTunneler
     Dns Domain:      domain.local
     Ip address:      192.168.11.13
     Network mask:    255.255.255.0
     Gateway     :    192.168.11.1
     Dns Server 1:    192.168.11.1_
     Dns Server 2:

                         < Continue >

(c) 2012 Virtual Cable S.L.
```

Step 2.- Fill in connection data to UDS Broker server:

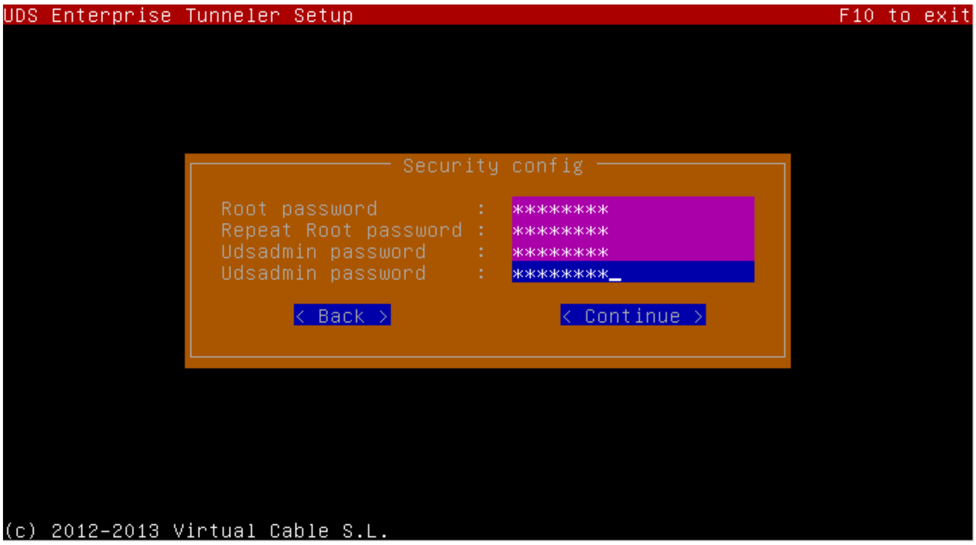
- **UDS Broker IP address.**
- **Communication Port with UDS Broker.**
- **Use of SSL connection (secure connection).**



```
UDS Enterprise Tunneler Setup                                        F10 to exit

                    Broker configuration

            Broker Server:        192.168.11.12
            Broker Port:          443_
            Use SSL:              [X]

                < Back >              < Continue >

(c) 2012 Virtual Cable S.L.
```

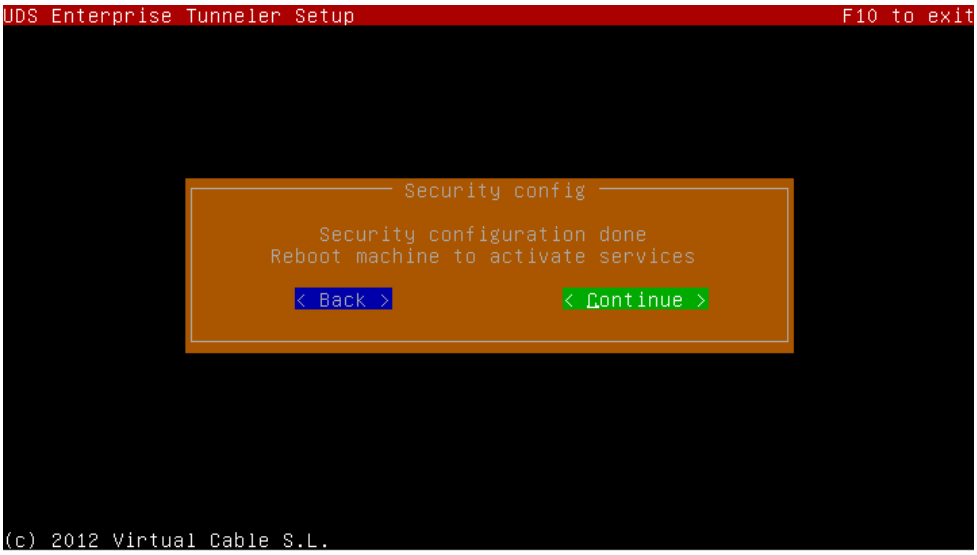
Step 3.- User accesses and UDS Tunneler server passwords configuration. Fill in the following data:

- **Root Password:** Root user password to enter UDS Tunneler server from VMware client console.
- **Udsadmin Password: Udsadmin** password that will allow to make a ssh connection to the UDS server machine.



```
UDS Enterprise Tunneler Setup                                        F10 to exit

                 Security config

   Root password          :  *********
   Repeat Root password   :  *********
   Udsadmin password      :  *********
   Udsadmin password      :  *********_

(c) 2012-2013 Virtual Cable S.L.
```

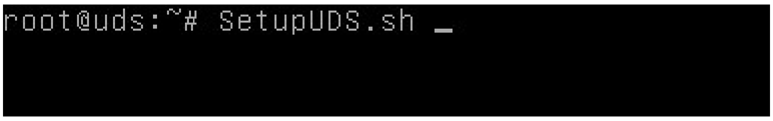
Finish and restart the virtual machine:



```
UDS Enterprise Tunneler Setup                                        F10 to exit

                 Security config

          Security configuration done
       Reboot machine to activate services

(c) 2012 Virtual Cable S.L.
```

Once restarted, the UDS tunneler is ready to use together with the UDS Broker.

If a rerun of the configuration wizard is needed, we will have to validate us in the server using the credentials by default and we run “SetupUDS.sh”.



```
root@uds:~# SetupUDS.sh _
```

Once the new data are entered, we will have to manually restart the server using “Reboot” command.

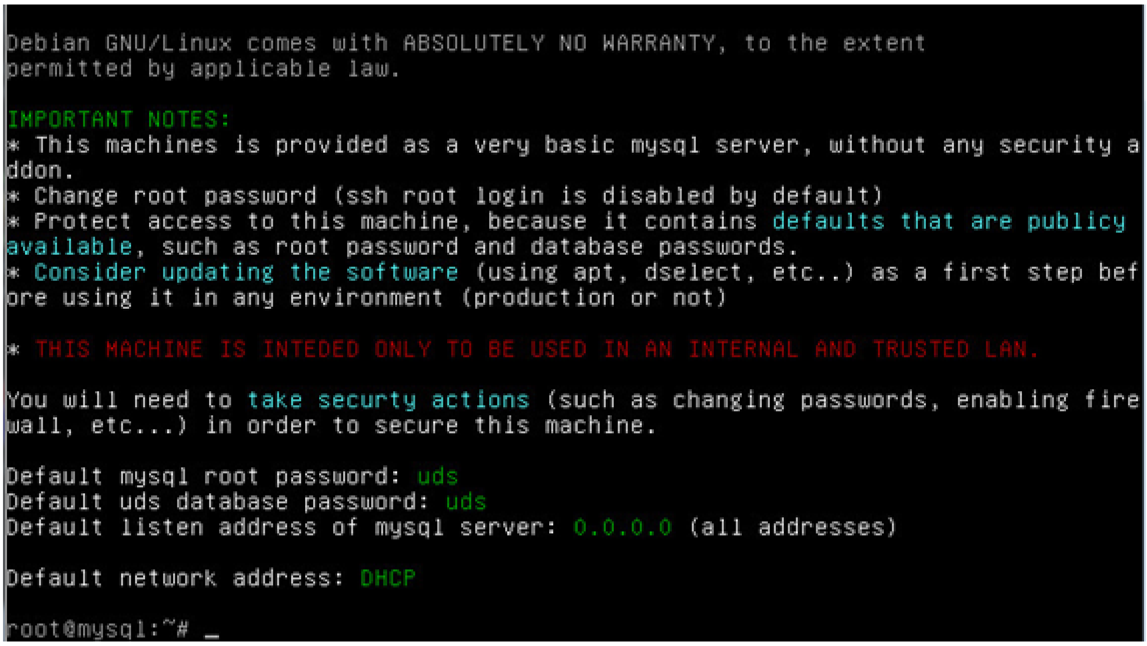
3.2.5 UDS database installation and configuration

In the event that VirtualCable provides the UDS database Virtual Appliance, you must follow the next steps:

Access the MySQL server with the following credentials:

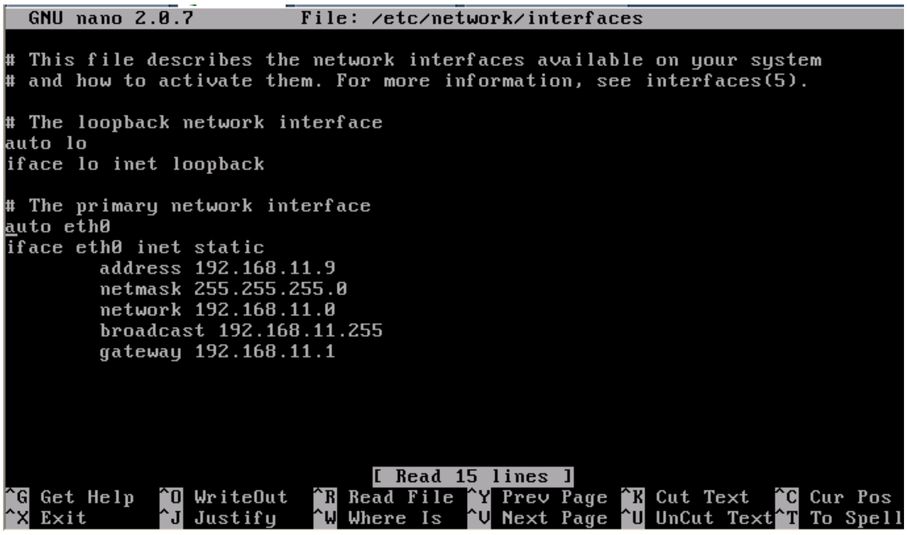
User: root

Password: uds



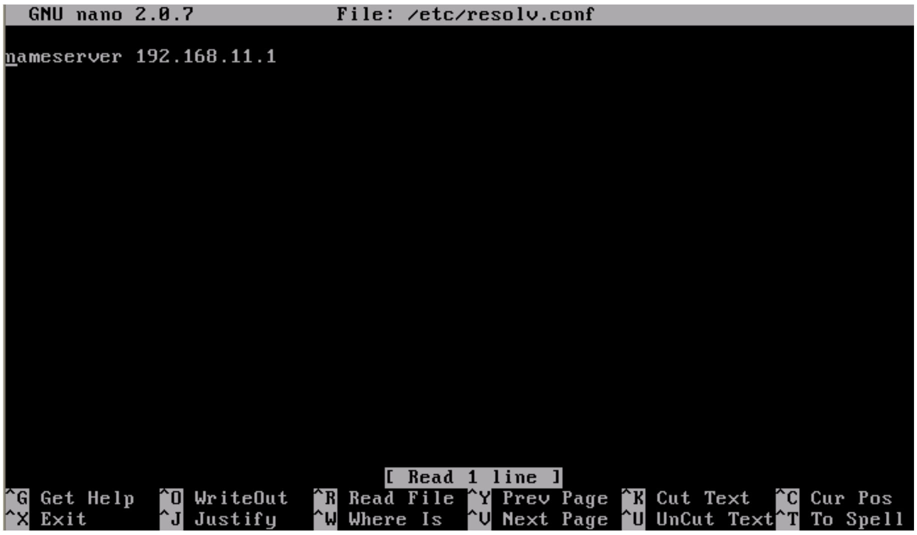
```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

IMPORTANT NOTES:
* This machines is provided as a very basic mysql server, without any security a
ddon.
* Change root password (ssh root login is disabled by default)
* Protect access to this machine, because it contains defaults that are publicy
available, such as root password and database passwords.
* Consider updating the software (using apt, dselect, etc..) as a first step bef
ore using it in any environment (production or not)

* THIS MACHINE IS INTEDED ONLY TO BE USED IN AN INTERNAL AND TRUSTED LAN.

You will need to take securty actions (such as changing passwords, enabling fire
wall, etc...) in order to secure this machine.

Default mysql root password: uds
Default uds database password: uds
Default listen address of mysql server: 0.0.0.0 (all addresses)

Default network address: DHCP

root@mysql:~# _
```

We configure the virtual machine network parameters. In order to do that, we modify the file ***"interfaces"*** and we assign it a static IP address (The Virtual Appliance is configured with dhcp by default)



```
  GNU nano 2.0.7           File: /etc/network/interfaces

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
        address 192.168.11.9
        netmask 255.255.255.0
        network 192.168.11.0
        broadcast 192.168.11.255
        gateway 192.168.11.1




                               [ Read 15 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

You also must modify the file ***"resolv.conf"*** to configure the dns server.



```
  GNU nano 2.0.7              File: /etc/resolv.conf

nameserver 192.168.11.1


                                [ Read 1 line ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

Once the IP data of the VM have been configured, it is already available for its use with UDS. By default, the MySQL server has configured the following DB ready to be used with the UDS server:

Instance: uds

User: uds

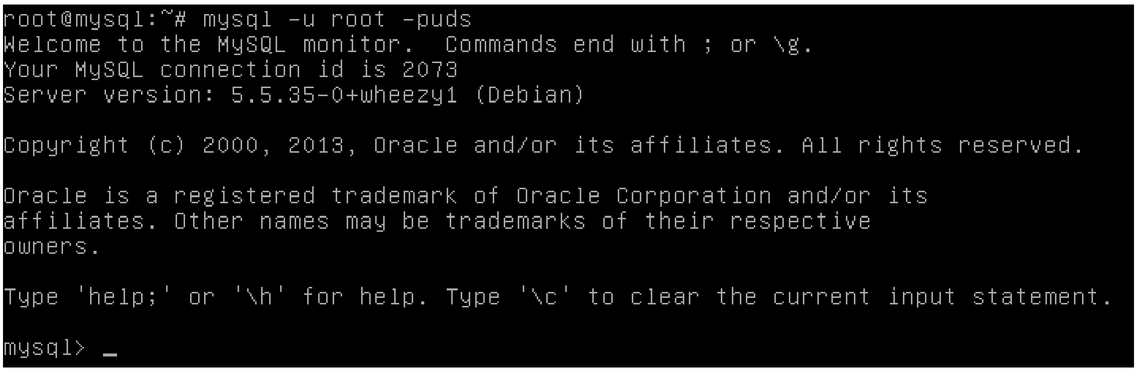
Password: uds

If you need to create a new DB instance for UDS, you should follow this process:

Access MySQL with the following credentials:

User: root

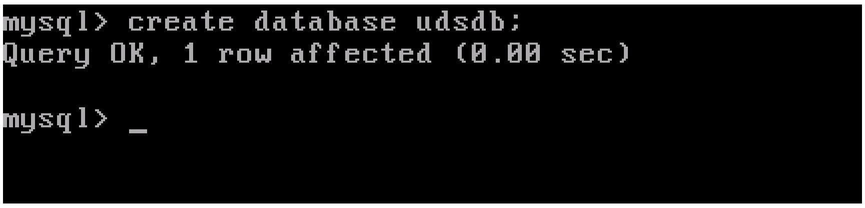
Password: uds



```
root@mysql:~# mysql -u root -puds
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2073
Server version: 5.5.35-0+wheezy1 (Debian)

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> _
```

The database is created by using the following command:

create database database_name;



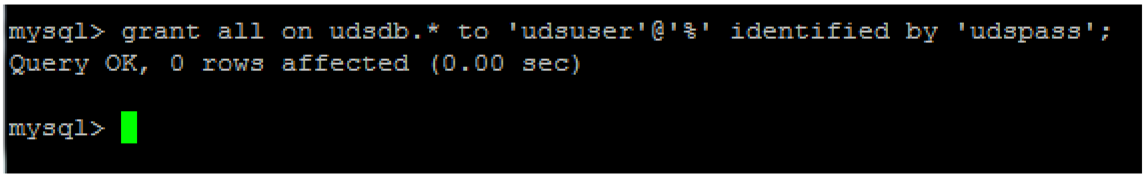
```
mysql> create database udsdb;
Query OK, 1 row affected (0.00 sec)

mysql> _
```

UTF-8 collation is assigned by the command:

alter database "database_name" character set "UTF8" collate "UTF8_general_ci";

A username with administrator permission is created in the new database by using the command:

grant all on database_name.* to 'usuario'@'%' identified by 'password';

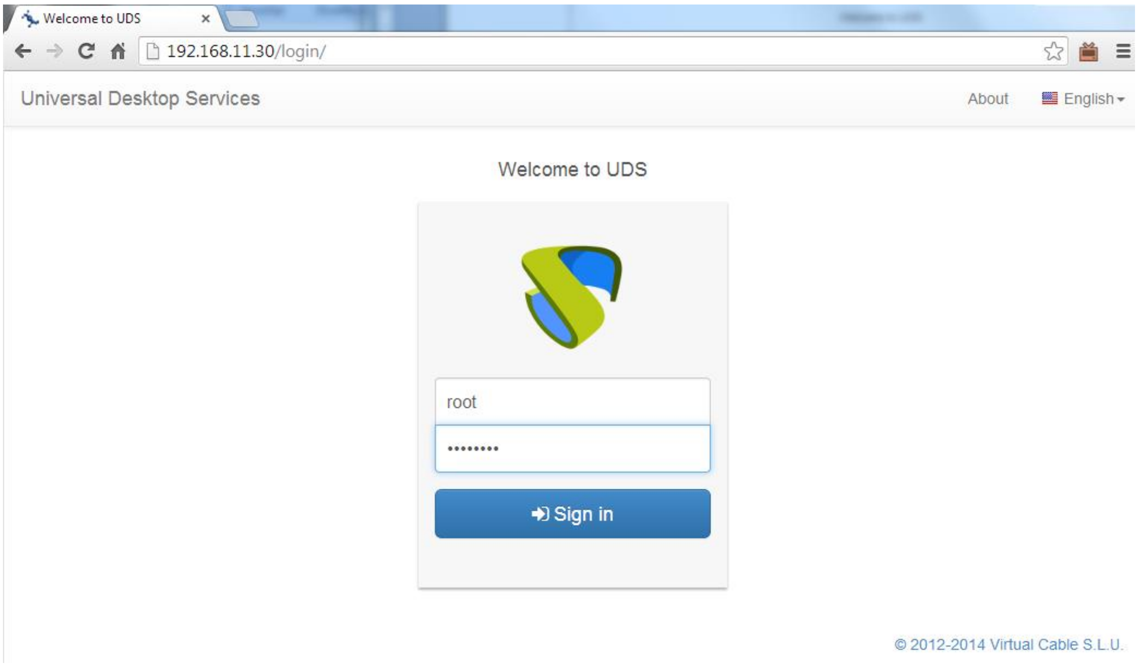


```
mysql> grant all on udsdb.* to 'udsuser'@'%' identified by 'udspass';
Query OK, 0 rows affected (0.00 sec)

mysql> 
```

The database will now be ready for use with the UDS system.

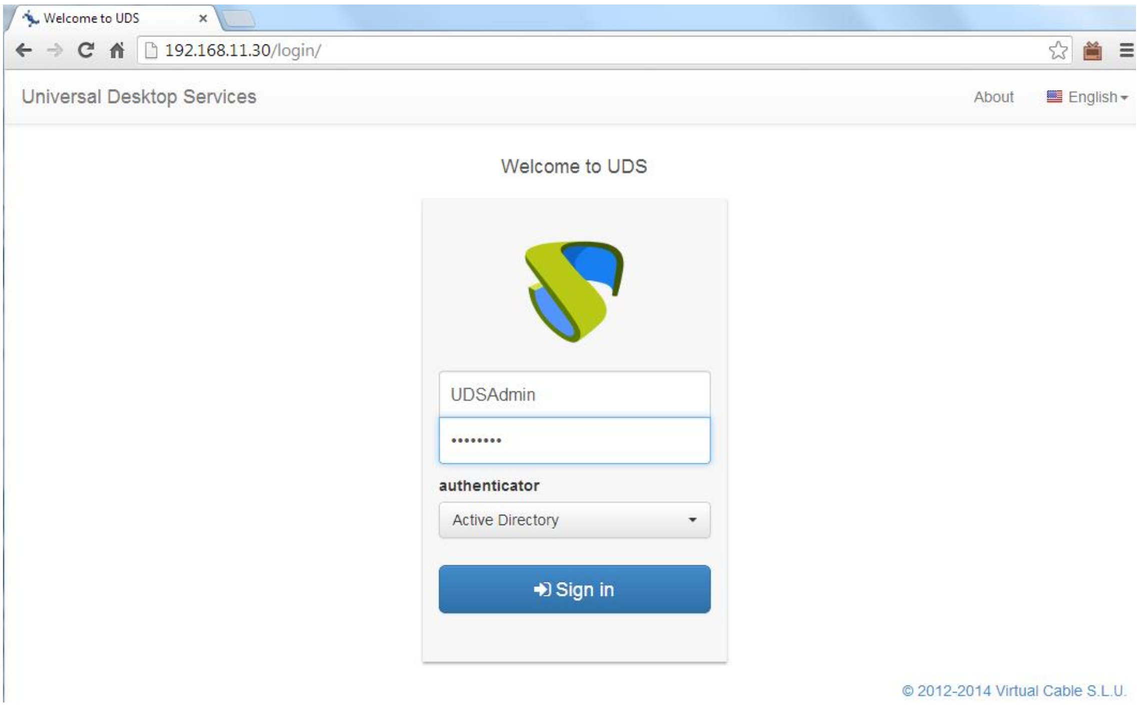
4 ADMINISTERING UDS - WEB ADMINISTRATION

Once the UDS platform has been installed, the system will be ready for its initial administering and configuration. To do this, enter the IP address or UDS Broker server name through http or https access.



The first time you enter UDS administration dashboard, you must enter using the "root" user and the password indicated in the UDS Broker Virtual Appliance configuration script (step 3.2.1). Once you access the administration dashboard, you will be able to change the password and create or select new users to enter administration dashboard.

If you already have an user with administration privileges for UDS platform, enter that user, password and select the authenticator that will validate the user.



192.168.11.30/login/

Welcome to UDS

UDSAdmin

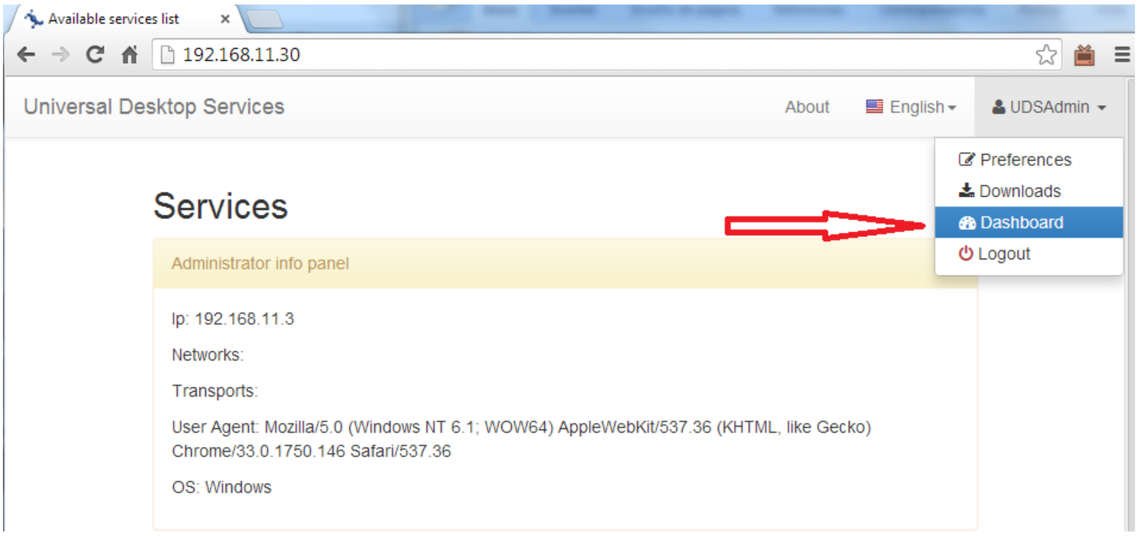
••••••••

authenticator

Active Directory

If we have more than one authenticator connected to UDS platform and we would like to access the administration dashboard with the Root user, the selected authenticator won't be used, because this user won't be validated against any authenticator.

In the user menu, select "Dashboard" to enter UDS administration:



Ip: 192.168.11.3

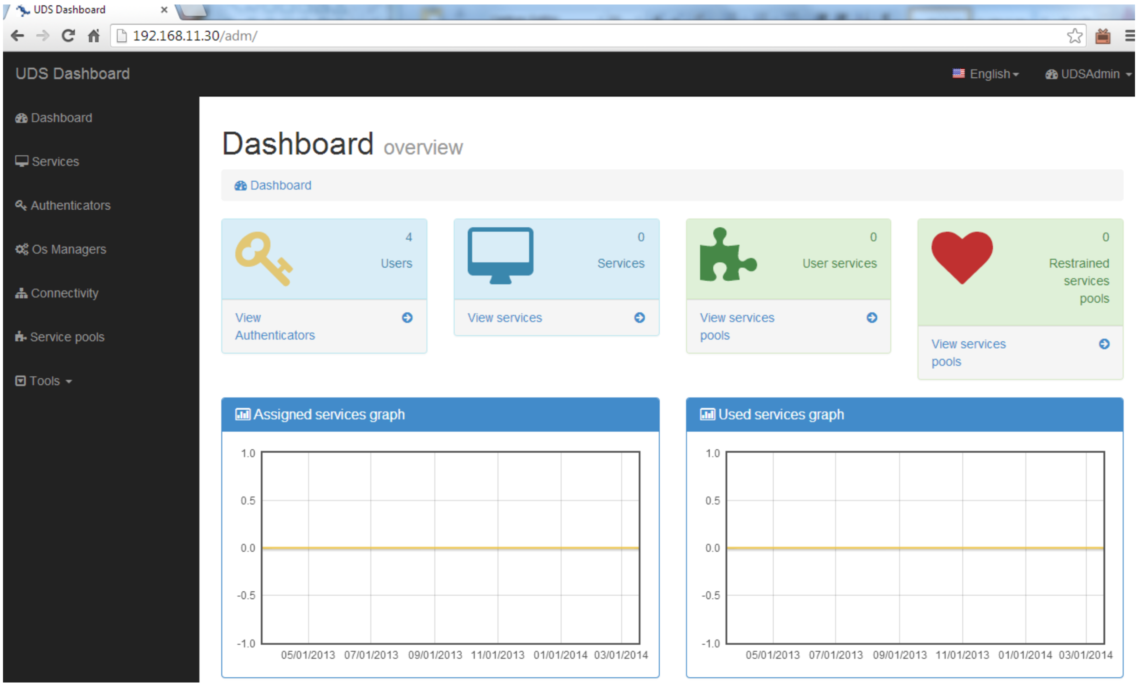
Networks:

Transports:

User Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/33.0.1750.146 Safari/537.36

OS: Windows

Once inside UDS administration, the initial configuration of the services will begin.



The configuration of each "Services Pools" must be approached like the building of a puzzle, made up of different elements:

- Each "Services Pool" is made up of different elements or pieces (Base Services, OS Managers, Transports and Authenticators).
- Once the elements of the first "Services Pool" have been configured, the creation thereof will begin, repeating the process with the next "Services Pool", if there is one.
- All of the configured "Services Pool" together will form the type of virtual desktop deployment managed by the UDS platform.

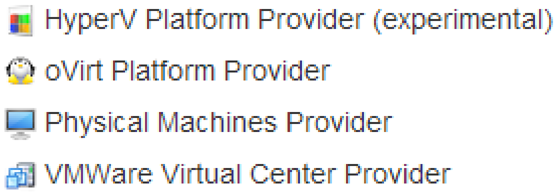
4.1 Configuring Service Providers

A Service Provider is the organization responsible for offering IP services.

The services offered by UDS will be on-demand virtual desktops provided by a virtualization platform or persistent physical/virtual desktops assigned to specific users via assignment of IPs.

In order to build a "Services Pool", it is necessary to at least have created a Service Provider.

Currently, UDS allows the following Service Providers:



HyperV Platform Provider (experimental)

oVirt Platform Provider

Physical Machines Provider

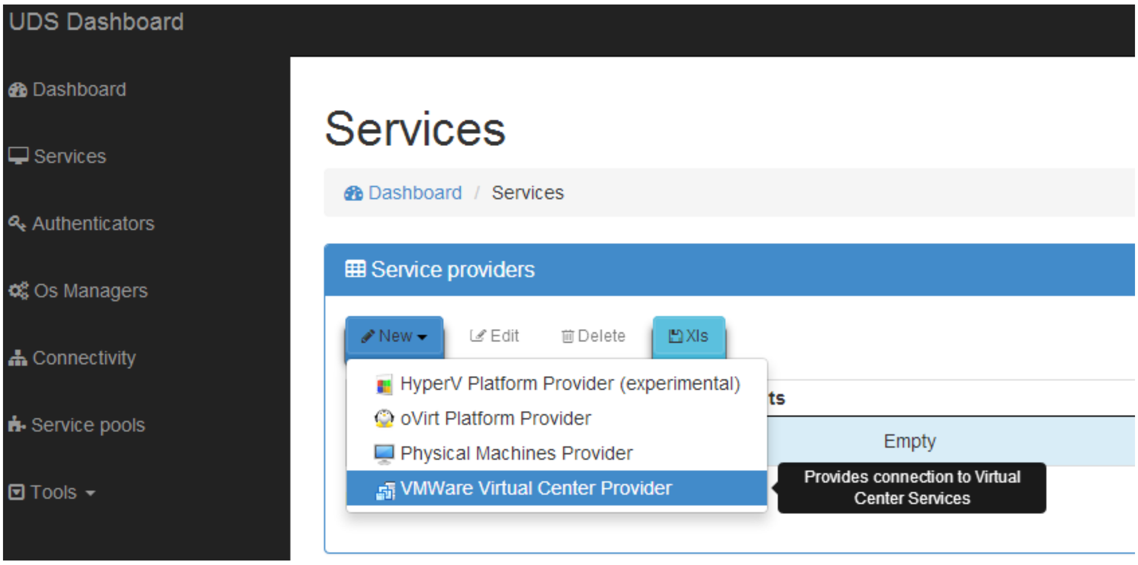
VMWare Virtual Center Provider

4.1.1 VDI platform with VMWare vSphere

Deployment of VDI platform via the VMware vSphere virtual infrastructure.

4.1.1.1 Registering “VMWare Virtual Center Provider” service provider

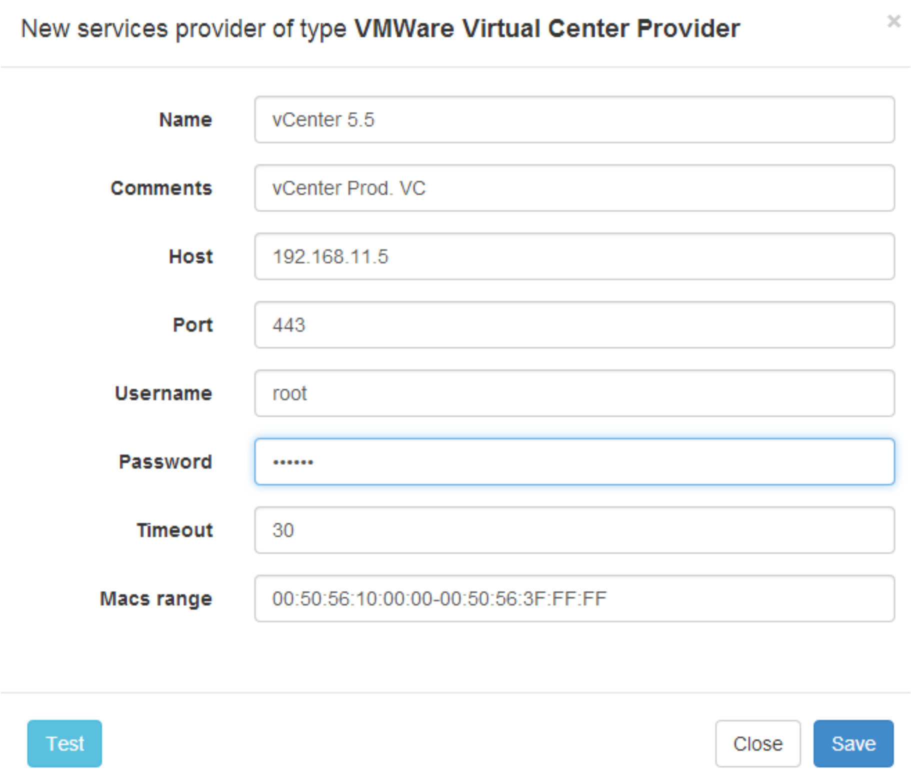
Enter "Services", click "New" and select “VMware Virtual Center Provider”.



In a VMware Virtual Center Provider, the minimum parameters to configure are: Service Name, vCenter server IP ("Host" field), an username and password with administrator rights on vCenter.

We can also select "Timeout" in the connection with vCenter and specify a range of MAC addresses for creating the virtual desktops.

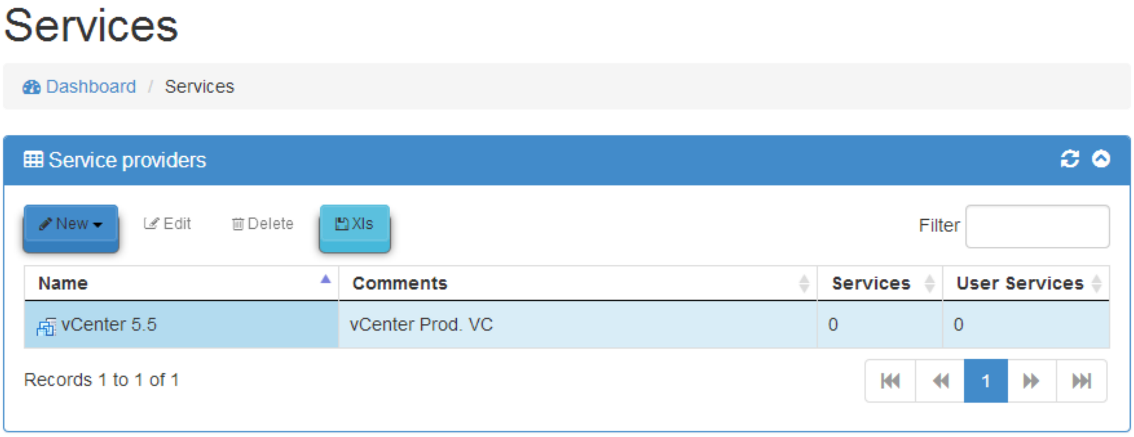
By clicking the "Test" button, we check if the connection has been made correctly.



New services provider of type **VMWare Virtual Center Provider**

Name	vCenter 5.5
Comments	vCenter Prod. VC
Host	192.168.11.5
Port	443
Username	root
Password	••••••
Timeout	30
Macs range	00:50:56:10:00:00-00:50:56:3F:FF:FF

When saving this configuration, we already have a valid "Service Providers" to start creating base services in the VMware vCenter platform. We can register all “VMware Virtual Center Provider” Service Providers we need in the UDS platform.



Services

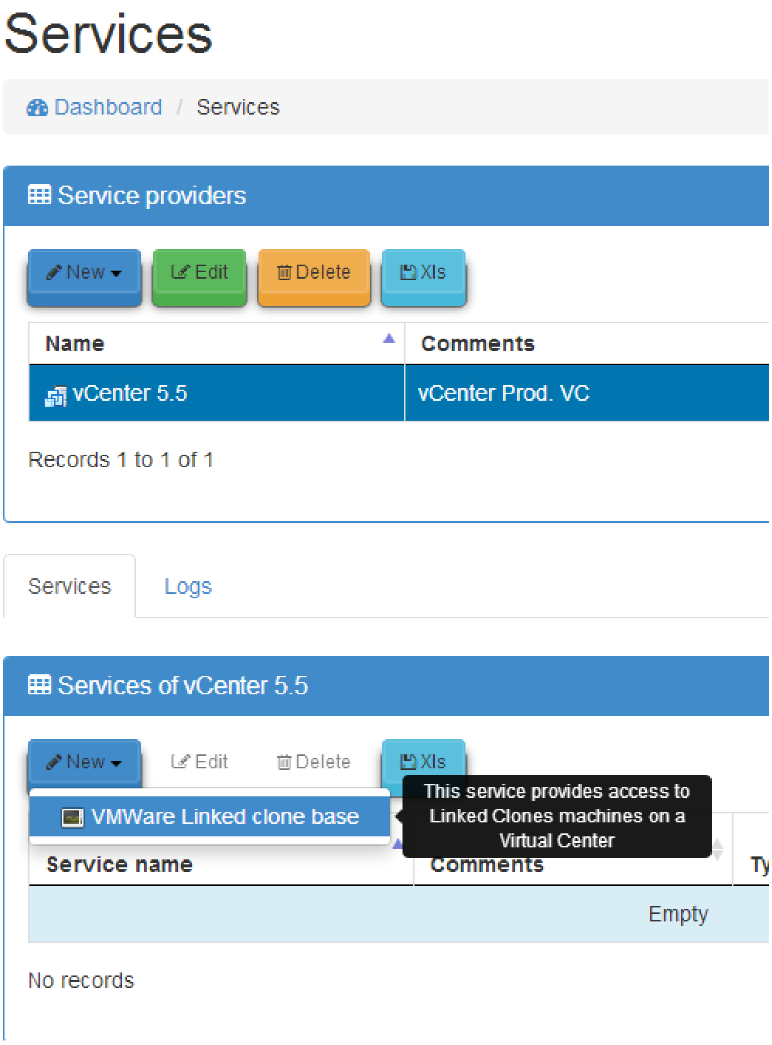
Name	Comments	Services	User Services
vCenter 5.5	vCenter Prod. VC	0	0

Records 1 to 1 of 1

4.1.1.2 Configuring service based on “VMWare Linked Clones”

Once the vSphere platform where the desktops will be created has been configured, you must create Base Services based on VMWare Linked Clones.

Select the Service Providers where we are going to create a VMware Linked Clone and click "New".



Name	Comments
vCenter 5.5	vCenter Prod. VC

Records 1 to 1 of 1

This service provides access to Linked Clones machines on a Virtual Center

Service name	Comments	Ty
	Empty	

No records

Choose a descriptive name for the template and configure the service parameters:

Datacenter: Datacenter where the service will be hosted.

Network: Network to which the desktops will be connected.

Pub. Resource Pool: vCenter resources Pool where the Linked Clones virtual desktops will be hosted (if there are no Pools in the VMware infrastructure, they will be created in the root).

Clones Folder: Location of the Linked Clones virtual desktops in the VMs view and the vCenter templates.

Resource Pool: vCenter resources pool where the template to be used by the service is located.

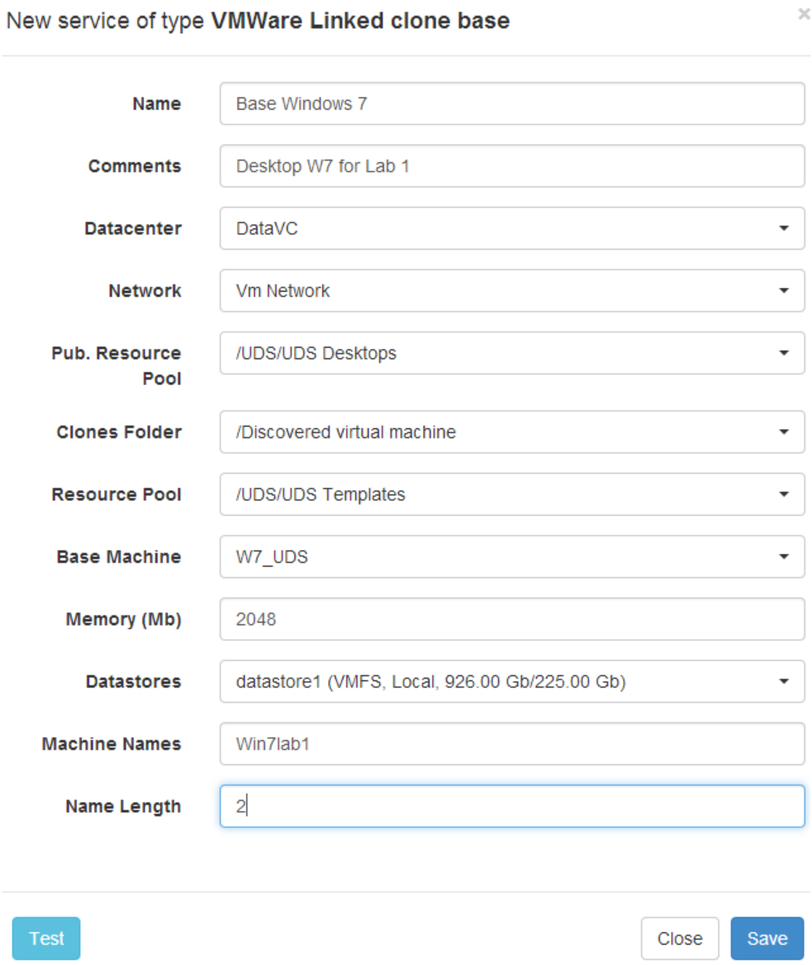
Base Machine: Template for deploying the virtual desktops.

Memory: Amount of memory to be assigned to the Linked Clones virtual desktops.

Datastores: Location where the publication of the service and the Linked Clones created will be stored. We can select one, several or all of the datastores clicking the "Ctrl" button. If you select more than one, the system will always locate the new publications and desktops in the Datastore with more free space (By default, the system won't generate new publications or new virtual desktops in datastores with less than 30GB of free space. This parameter can be modified entering the UDS system advanced options).

Machine Names: Root name of all of the Linked Clones virtual desktops to be deployed on this service. (ex: Machine Names= Win7lab1).

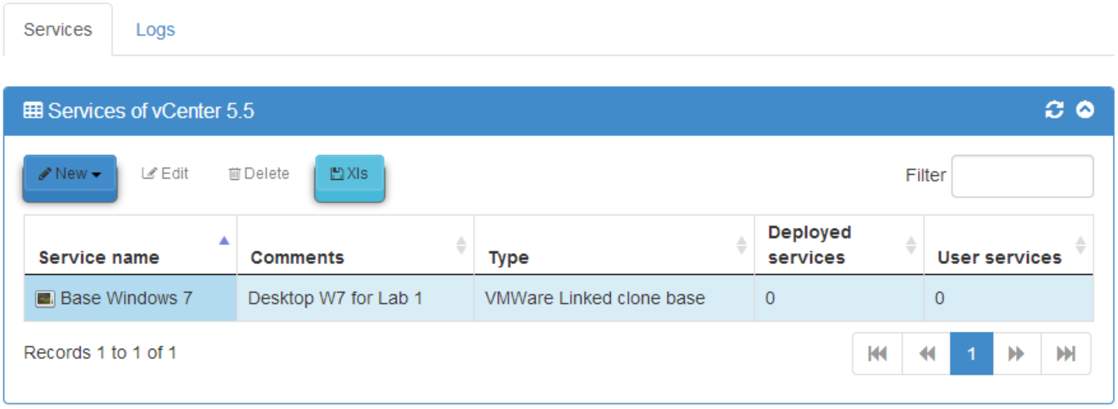
Name Length: Number of digits of the counter attached to the root name of the desktops (ex: Name Length= 2, Win7lab1**01**...Win7lab1**99**).



New service of type **VMWare Linked clone base**

Name	Base Windows 7
Comments	Desktop W7 for Lab 1
Datacenter	DataVC
Network	Vm Network
Pub. Resource Pool	/UDS/UDS Desktops
Clones Folder	/Discovered virtual machine
Resource Pool	/UDS/UDS Templates
Base Machine	W7_UDS
Memory (Mb)	2048
Datastores	datastore1 (VMFS, Local, 926.00 Gb/225.00 Gb)
Machine Names	Win7lab1
Name Length	2

When saving this configuration, we already have a valid "VMware Linked Clone Base" in the VMware vCenter platform. We can register all “VMware Linked Clone Base" we need in the UDS platform.

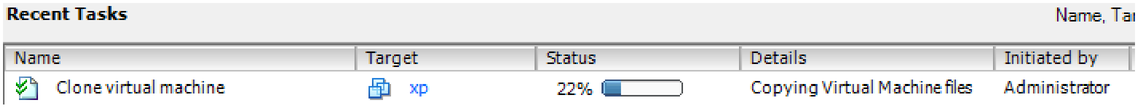


Service name	Comments	Type	Deployed services	User services
Base Windows 7	Desktop W7 for Lab 1	VMWare Linked clone base	0	0

Records 1 to 1 of 1

Once the entire UDS environment has been configured and the first Service Pools has been created, we will be able to observe how the virtual desktops based on VMWare Linked Clones are deployed on the vCenter server.

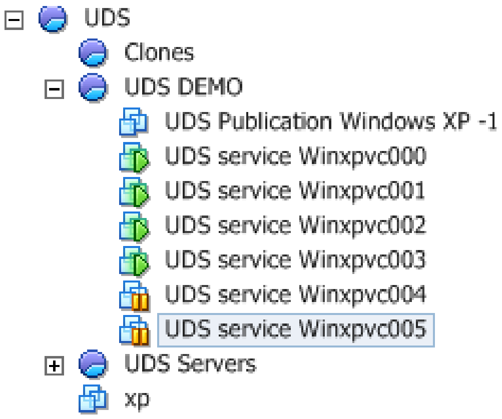
The first task that the vCenter will perform will be to create a base machine (this machine will be created each time we make a publication of a service) which will be a clone of the template selected when registering the service, with a hard drive size and characteristics equal to those of said template.



Name	Target	Status	Details	Initiated by
Clone virtual machine	xp	22%	Copying Virtual Machine files	Administrator

Once the process of creating the base machine has been completed (the UDS system calls it: "UDS Publication name_service –number_publication"), the creation of virtual desktops in the vCenter automatically begins (the UDS system calls it: "UDS service Machine_Name+Name_Length".

The hard drive space taken up by the virtual desktops ("Linked Clones") will be exclusively that which is taken up by the changes made by the users on the machines after their initial connection.



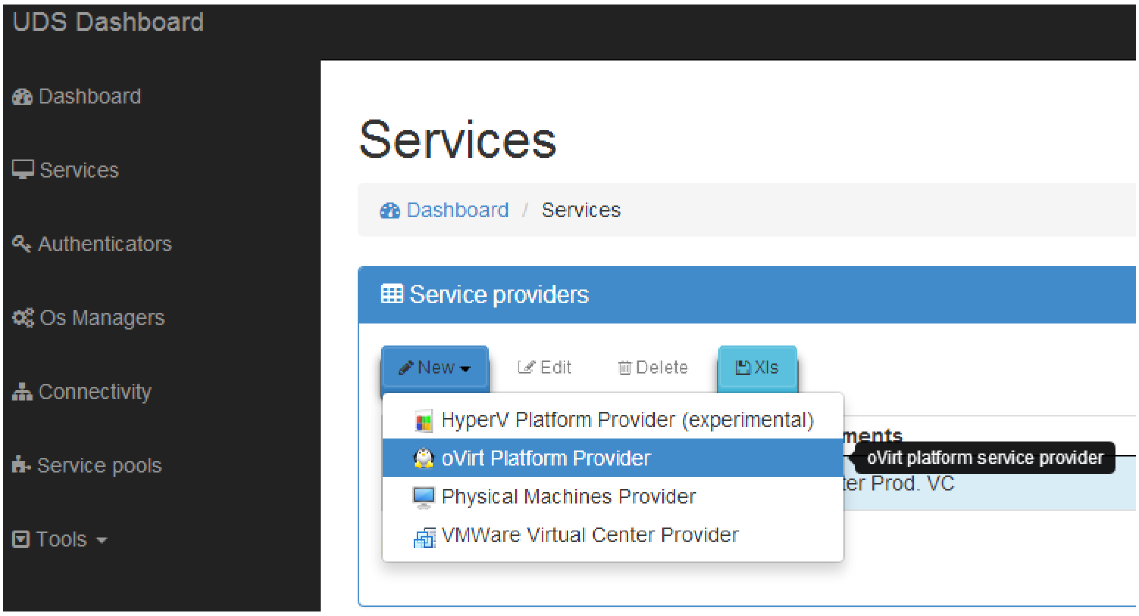
```
⊟ UDS
    Clones
  ⊟ UDS DEMO
      UDS Publication Windows XP -1
      UDS service Winxpvc000
      UDS service Winxpvc001
      UDS service Winxpvc002
      UDS service Winxpvc003
      UDS service Winxpvc004
      UDS service Winxpvc005
  ⊞ UDS Servers
    xp
```

4.1.2 VDI platform with oVirt

Deploying the VDI platform via the virtual oVirt infrastructure.

4.1.2.1 Registration of service provider "ovirt Platform Provider"

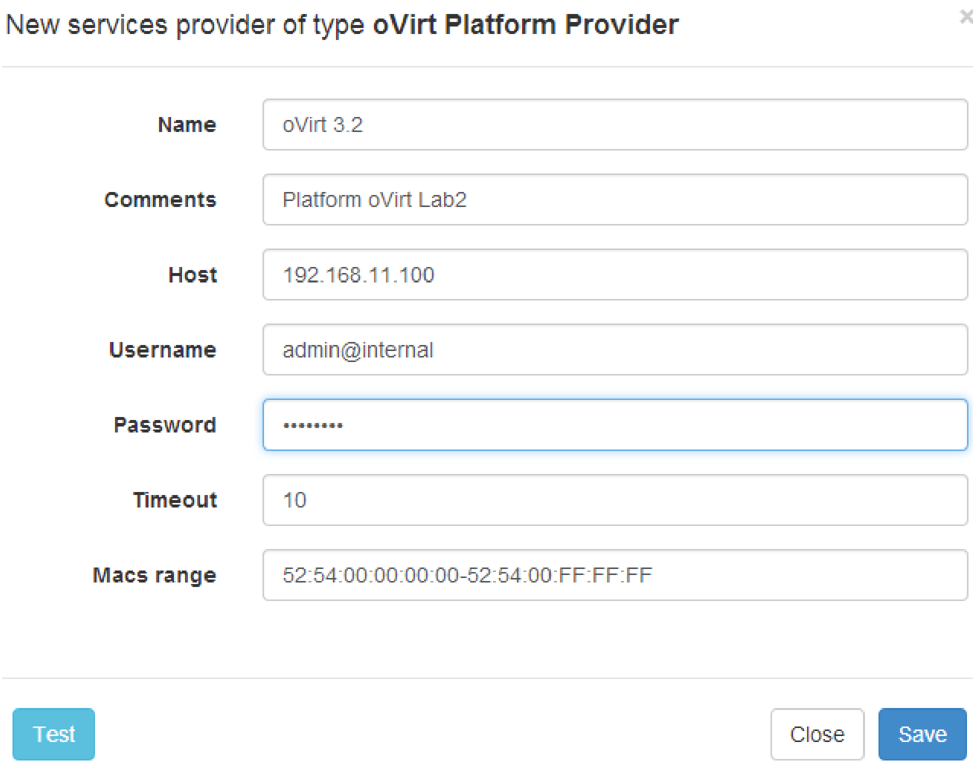
Enter "Services", click "New" and select "oVirt Platform Provider":



In an "oVirt Platform Provider," you must configure at least the following parameters: Service Name, oVirt-engine IP server ("Host" field), username (with user@domain format) and password with administration rights on the oVirt-engine.

We can also indicate the "Timeout" in the connection with oVirt-engine and specify a range of MAC addresses for creating the virtual desktops.

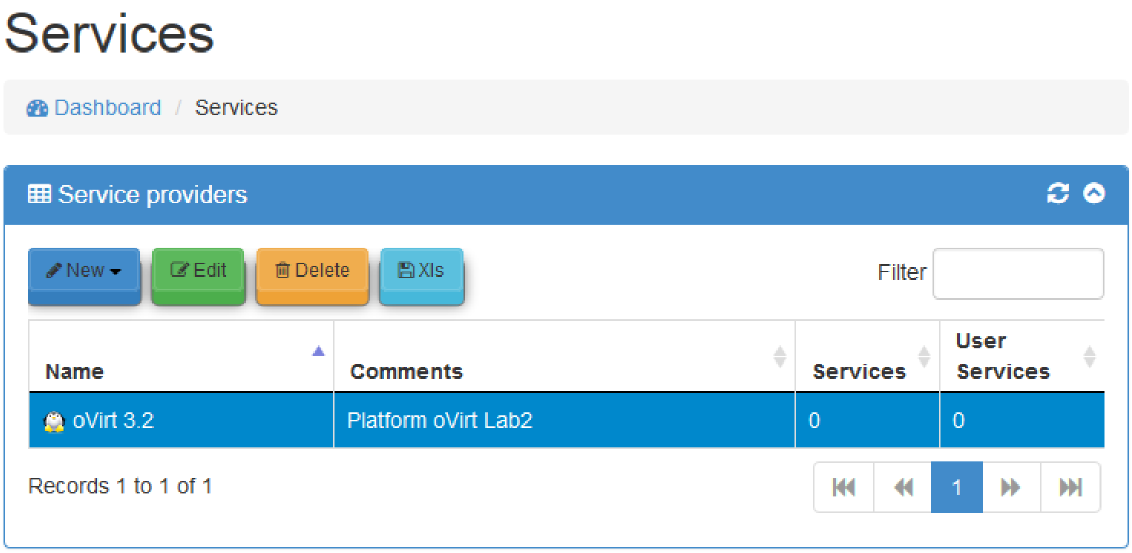
We will check that the connection has been correctly made by clicking on the "Test" button.



New services provider of type **oVirt Platform Provider**

Name	oVirt 3.2
Comments	Platform oVirt Lab2
Host	192.168.11.100
Username	admin@internal
Password	••••••••
Timeout	10
Macs range	52:54:00:00:00:00-52:54:00:FF:FF:FF

When saving this configuration, we already have a valid "Service Providers" to start creating base services in the oVirt platform. We can register all "oVirt Platform Provider"   Service Providers we need in the UDS platform.



Services

Dashboard / Services

Service providers

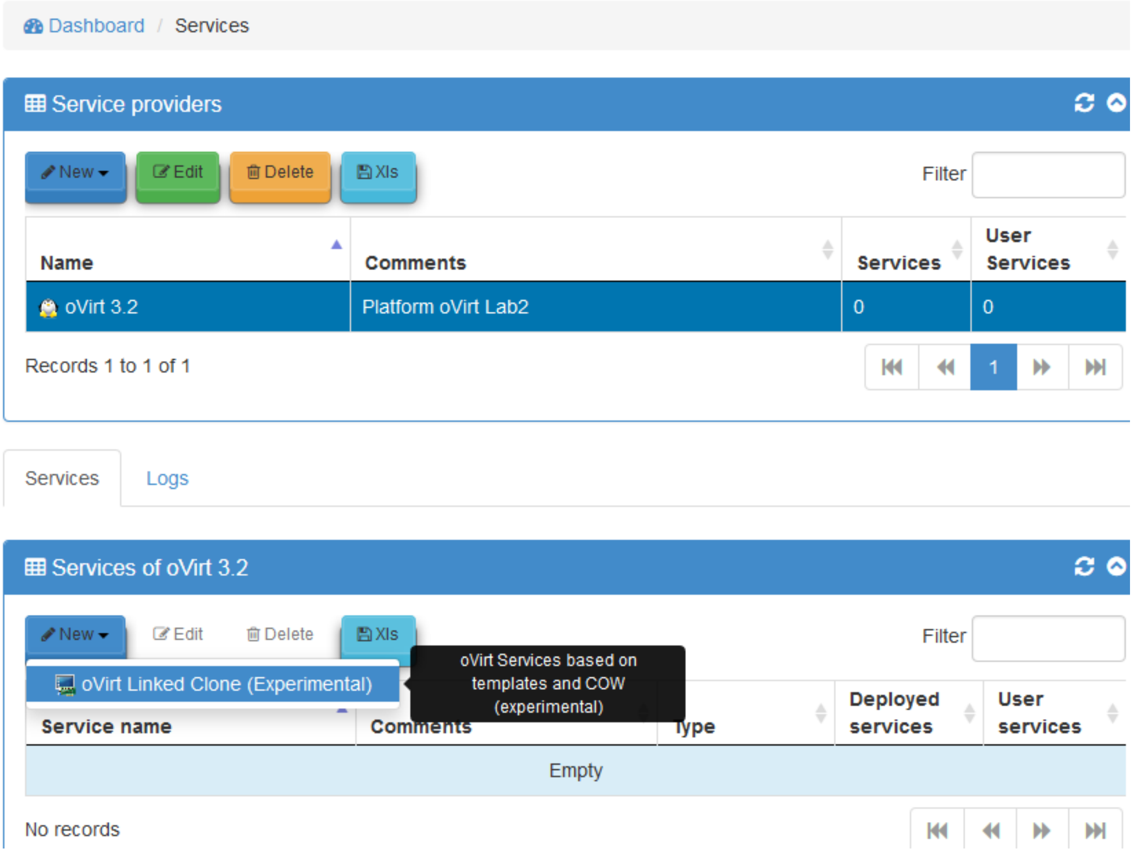
Name	Comments	Services	User Services
oVirt 3.2	Platform oVirt Lab2	0	0

Records 1 to 1 of 1

4.1.2.2 Configuring service based on "oVirt Linked Clone"

Once the oVirt platform where the desktops will be created has been configured, you must create base services based on oVirt Linked Clones.

Select the Service Providers where we are going to create an oVirt Linked Clone and click "New".



Name	Comments	Services	User Services
oVirt 3.2	Platform oVirt Lab2	0	0

Records 1 to 1 of 1

oVirt Linked Clone (Experimental)

oVirt Services based on templates and COW (experimental)

Service name	Comments	Type	Deployed services	User services
Empty				

No records

Type a descriptive name for the template and configure the service parameters:

Base Machine: Template for deploying the virtual desktops.

Cluster: oVirt node cluster that will host the deployed Linked Clones.

Datastore Domain: Storage established for deploying the Linked Clones.

Reserved Space: Minimun free space a Datastore may have to be used by UDS system.

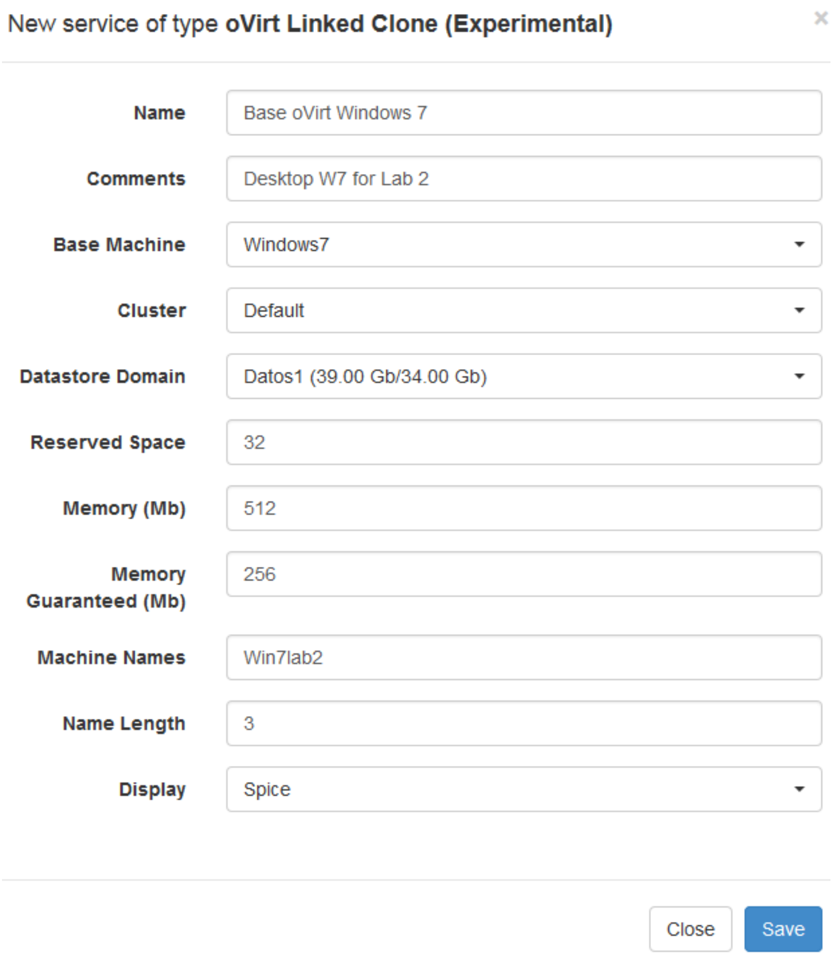
Memory: Amount of memory that will be assigned to the Linked Clones.

Memory Guaranteed: Amount of memory that will be guaranteed to the Linked Clones.

Machine Names: Root name of all of the Linked Clones to be deployed in this service (ex. Machine Names= Win7lab2).

Name Length: Number of counter digits attached to the root name of the desktops (ex: Name Length= 3, Win7lab2**001**… Win7lab2**999**).

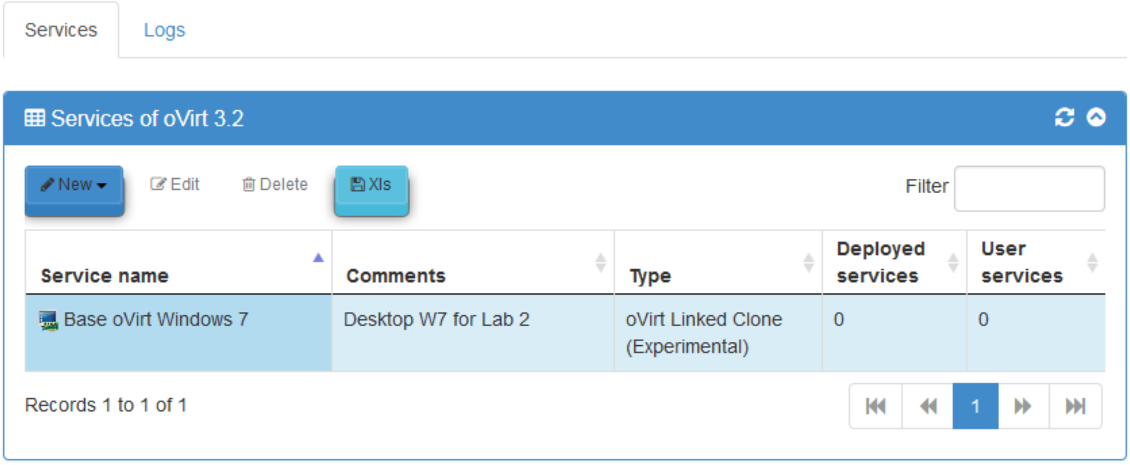
Display: Connection protocol of the virtual desktops deployed via Linked Clones.



New service of type **oVirt Linked Clone (Experimental)**

Name	Base oVirt Windows 7
Comments	Desktop W7 for Lab 2
Base Machine	Windows7
Cluster	Default
Datastore Domain	Datos1 (39.00 Gb/34.00 Gb)
Reserved Space	32
Memory (Mb)	512
Memory Guaranteed (Mb)	256
Machine Names	Win7lab2
Name Length	3
Display	Spice

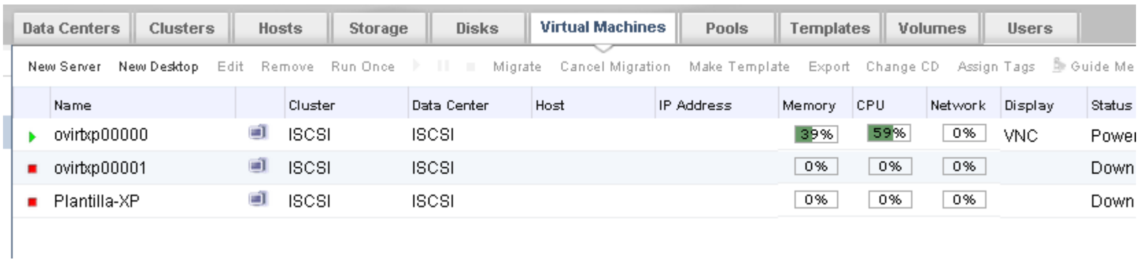
When saving this configuration, we already have a valid "oVirt Linked Clone" in the oVirt platform. We can register all "oVirt Linked Clone" we need in the UDS platform.



Service name	Comments	Type	Deployed services	User services
Base oVirt Windows 7	Desktop W7 for Lab 2	oVirt Linked Clone (Experimental)	0	0

Records 1 to 1 of 1

When the entire UDS environment has been configured and the first deployed service has been created, we will be able to observe how the virtual desktops based on oVirt Linked Clones are deployed on the oVirt-engine server.



Name	Cluster	Data Center	Host	IP Address	Memory	CPU	Network	Display	Status
ovirtxp00000	ISCSI	ISCSI			39%	59%	0%	VNC	Powei
ovirtxp00001	ISCSI	ISCSI			0%	0%	0%		Down
Plantilla-XP	ISCSI	ISCSI			0%	0%	0%		Down

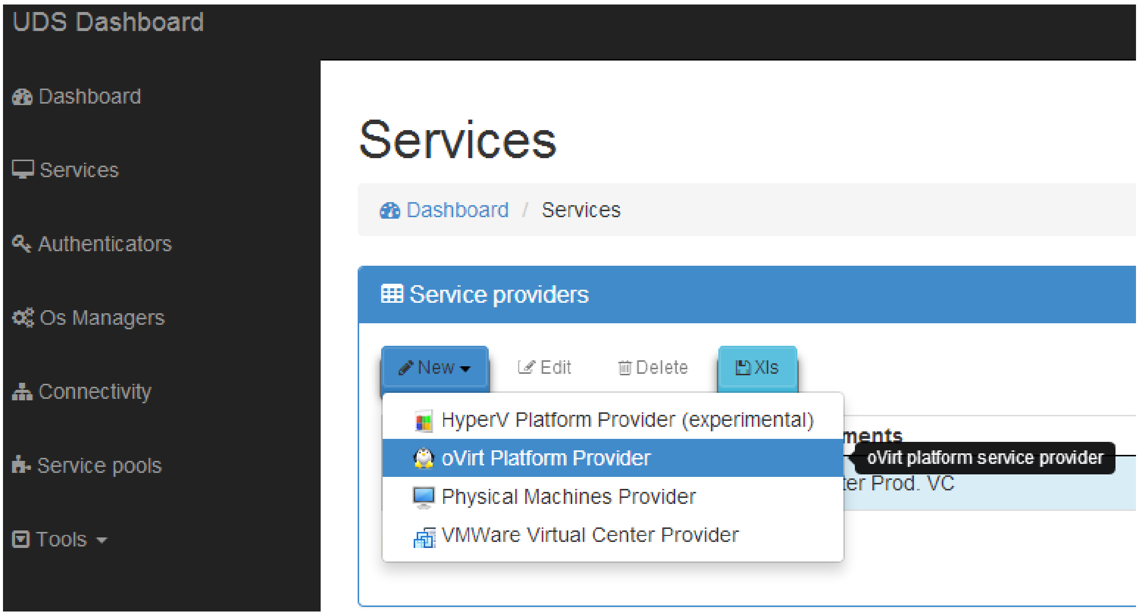
4.1.3 VDI platform with RHEV

Deploying the VDI platform via virtual Red Hat Enterprise Virtualization infrastructure.

4.1.3.1 Registration of service provider "ovirt Platform Provider"

To connect a RHEV platform in UDS, we must use "oVirt Platform Provider" service provider.

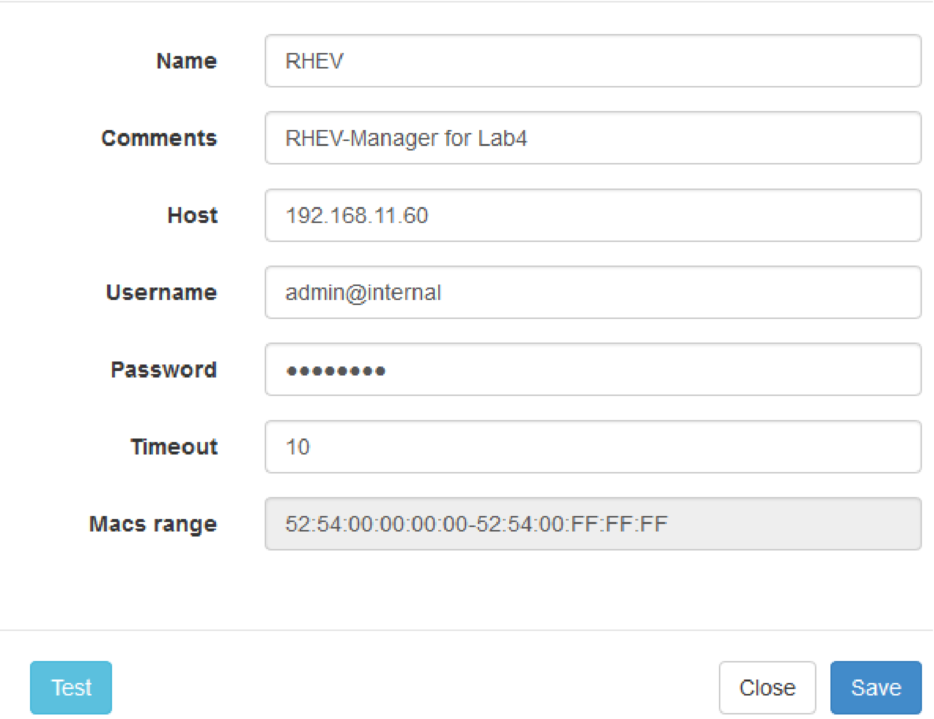
Enter "Services", click "New" and select "oVirt Platform Provider":



When we use an "oVirt Platform Provider" to connect a RHEV platform you must configure at least the following parameters: Service Name, RHEV-Manager IP server ("Host" field), username (with user@domain format) and password with administration rights on RHEV-Manager.

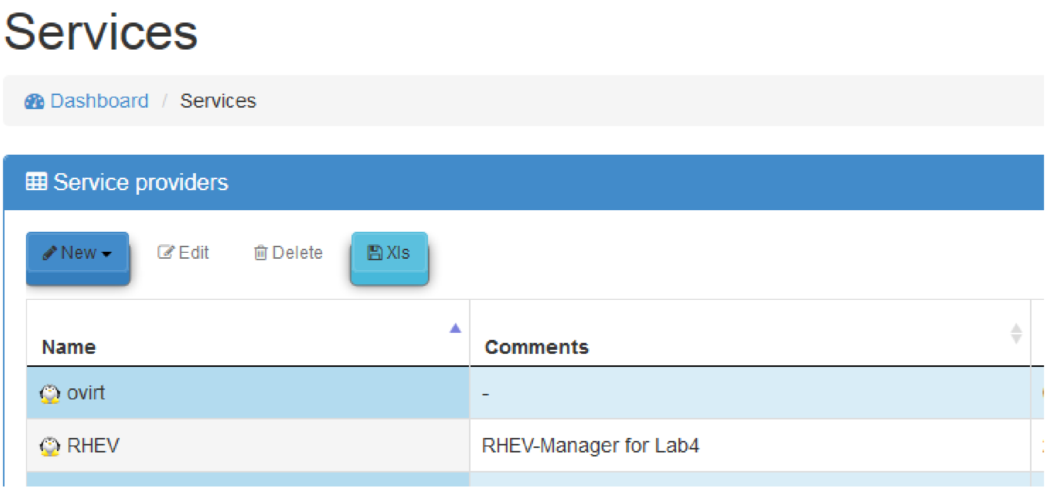
We can also indicate the "Timeout" in the connection with RHEV-Manager and specify a range of MAC addresses to create the virtual desktops.

We will check that the connection has been correctly made by clicking on the "Test" button.



Name	RHEV
Comments	RHEV-Manager for Lab4
Host	192.168.11.60
Username	admin@internal
Password	••••••••
Timeout	10
Macs range	52:54:00:00:00:00-52:54:00:FF:FF:FF

When saving this configuration, we already have a valid "Service Providers" to start creating base services in the RHEV platform. We can register all "oVirt Platform Provider"   Service Providers we need in the UDS platform.



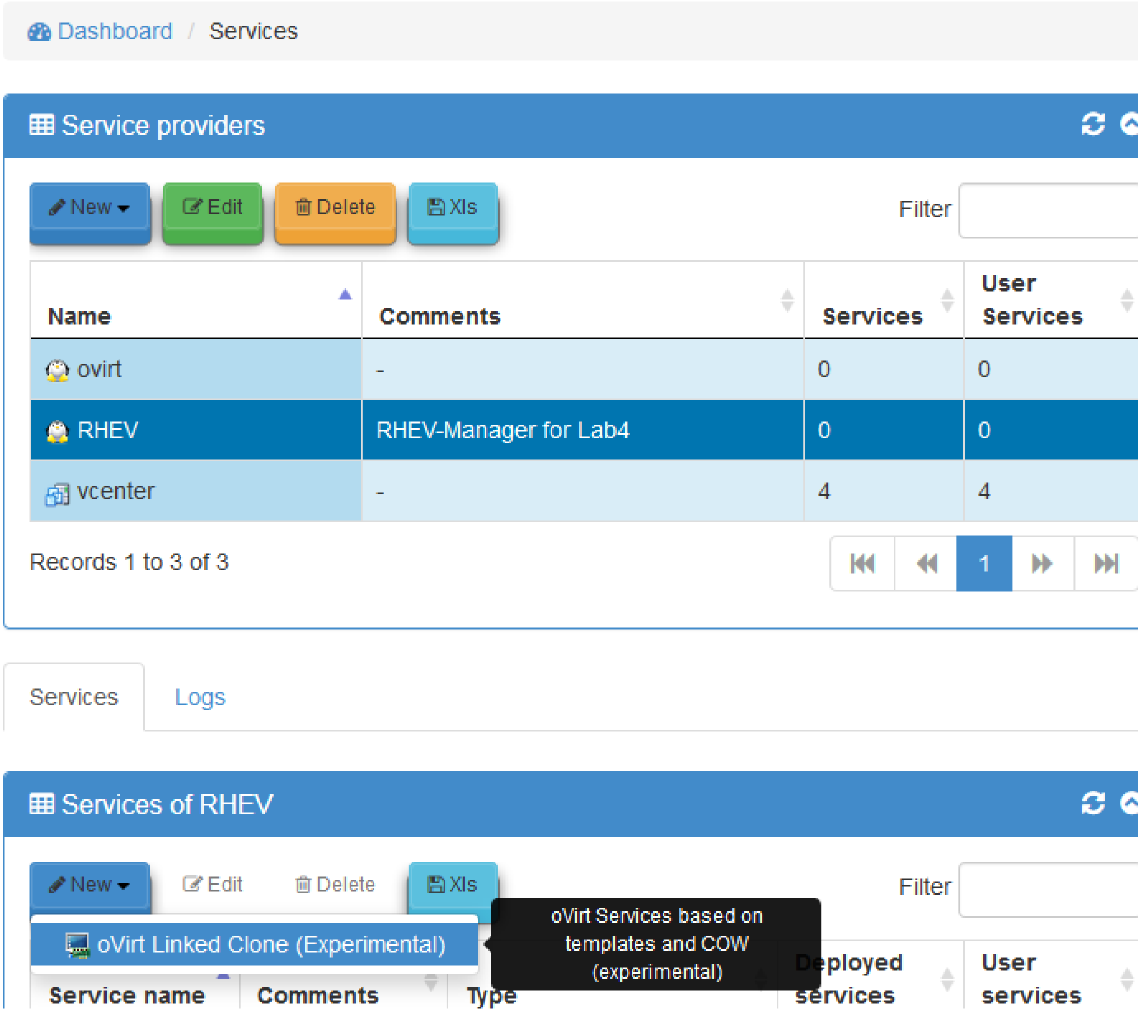
Services

Name	Comments
ovirt	-
RHEV	RHEV-Manager for Lab4

4.1.3.2 Configuring service based on “oVirt Linked Clone”

Once the RHEV platform through “oVirt Platform Provider” service provider, where the desktops will be created, has been configured, you must create base services based on oVirt Linked Clones.

Select the Service Providers where we are going to create an oVirt Linked Clone and click "New".



Name	Comments	Services	User Services
ovirt	-	0	0
RHEV	RHEV-Manager for Lab4	0	0
vcenter	-	4	4

Records 1 to 3 of 3

oVirt Services based on templates and COW (experimental)

Type a descriptive name for the template and configure the service parameters:

Base Machine: Virtual Machine Image the Linked Clokes will be deployed from.

Cluster: RHEV cluster that will host the deployed Linked Clones.

Datastore Domain: Storage established for deploying the Linked Clones.

Reserved Space: Minimun free space a Datastore may have to be used by UDS system.

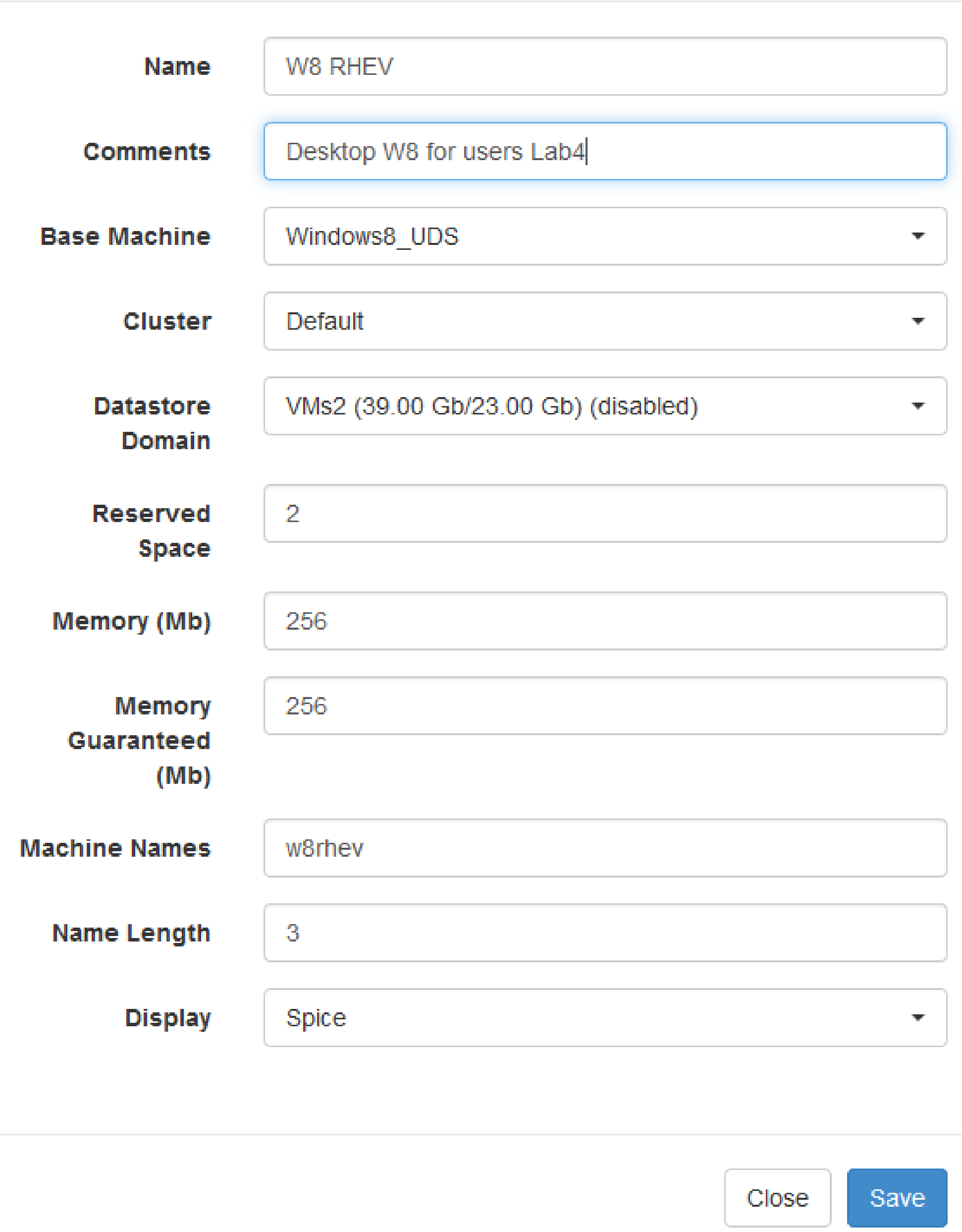
Memory: Amount of memory that will be assigned to the Linked Clones.

Memory Guaranteed: Amount of memory that will be guaranteed to the Linked Clones.

Machine Names: Root name of all of the Linked Clones to be deployed in this service (ex. Machine Names= Win7lab2).

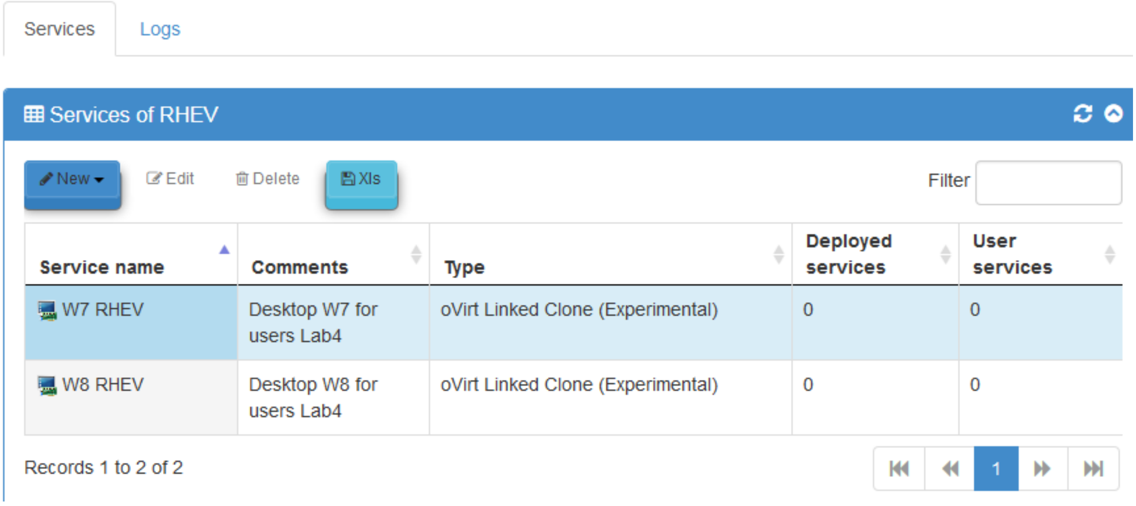
Name Length: Number of counter digits attached to the root name of the desktops (ex: Name Length= 3, Win7lab2001... Win7lab2999).

Display: Connection protocol of the virtual desktops deployed via Linked Clones.



Name	W8 RHEV
Comments	Desktop W8 for users Lab4
Base Machine	Windows8_UDS
Cluster	Default
Datastore Domain	VMs2 (39.00 Gb/23.00 Gb) (disabled)
Reserved Space	2
Memory (Mb)	256
Memory Guaranteed (Mb)	256
Machine Names	w8rhev
Name Length	3
Display	Spice

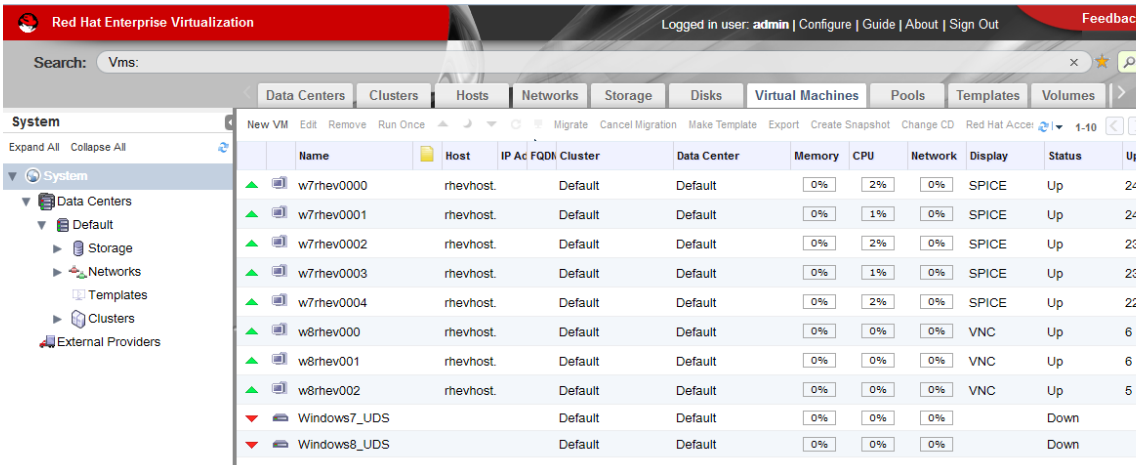
When saving this configuration, we already have a valid "oVirt Linked Clone" in RHEV platform. We can register all "oVirt Linked Clone" we need in the UDS platform.



Service name	Comments	Type	Deployed services	User services
W7 RHEV	Desktop W7 for users Lab4	oVirt Linked Clone (Experimental)	0	0
W8 RHEV	Desktop W8 for users Lab4	oVirt Linked Clone (Experimental)	0	0

Records 1 to 2 of 2

When the entire UDS environment has been configured and the first Service Pools has been created, we will be able to observe how the virtual desktops based on oVirt Linked Clones are deployed on the RHEV-Manager server.



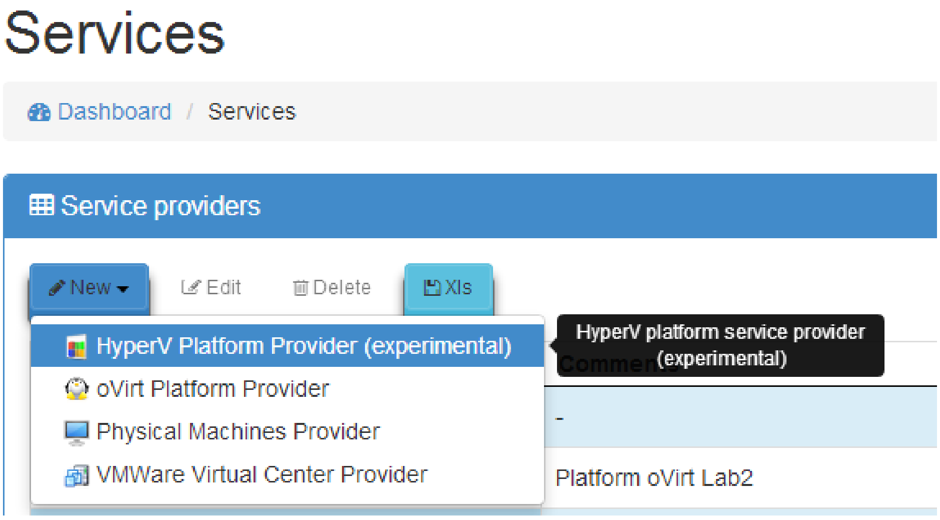
Name	Host	IP Ac	FQDN	Cluster	Data Center	Memory	CPU	Network	Display	Status	Up
w7rhev0000	rhevhost.			Default	Default	0%	2%	0%	SPICE	Up	24
w7rhev0001	rhevhost.			Default	Default	0%	1%	0%	SPICE	Up	24
w7rhev0002	rhevhost.			Default	Default	0%	2%	0%	SPICE	Up	23
w7rhev0003	rhevhost.			Default	Default	0%	1%	0%	SPICE	Up	23
w7rhev0004	rhevhost.			Default	Default	0%	2%	0%	SPICE	Up	22
w8rhev000	rhevhost.			Default	Default	0%	0%	0%	VNC	Up	6
w8rhev001	rhevhost.			Default	Default	0%	0%	0%	VNC	Up	6
w8rhev002	rhevhost.			Default	Default	0%	0%	0%	VNC	Up	5
Windows7_UDS				Default	Default	0%	0%	0%		Down	
Windows8_UDS				Default	Default	0%	0%	0%		Down	

4.1.4 VDI platform with Microsoft Hyper-V

Deploying the VDI platform via the virtual Microsoft Hyper-V infrastructure.

4.1.4.1 Registration of service provider “Hyper-V Platform Provider”

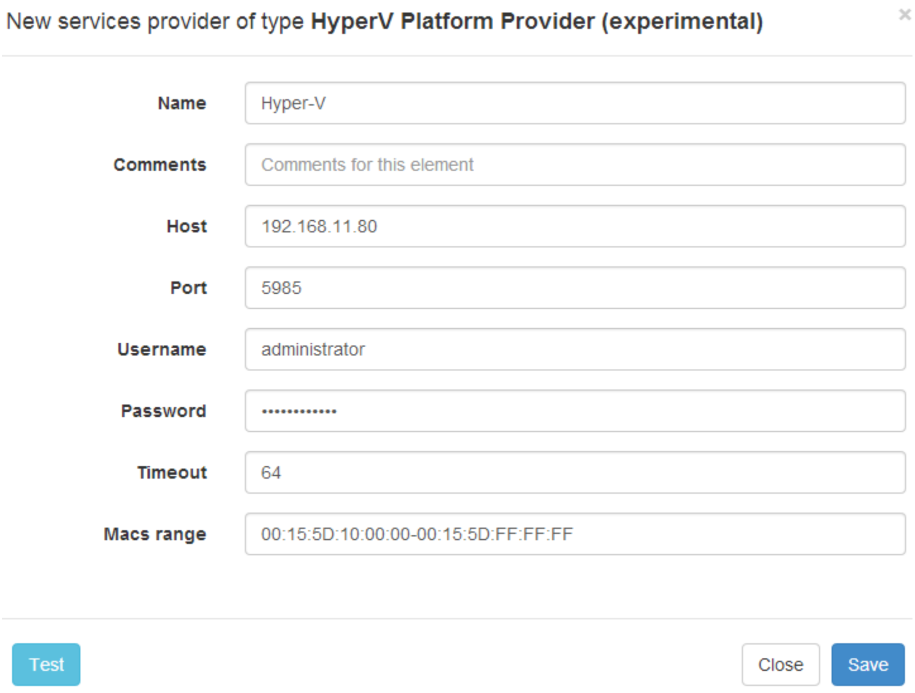
Enter "Services", click "New" and select “Hyper-V Platform Provider”.


Services
Dashboard / Services
Service providers
New
Edit
Delete
Xls
HyperV Platform Provider (experimental)
oVirt Platform Provider
Physical Machines Provider
VMWare Virtual Center Provider
HyperV platform service provider (experimental)
Comment
-
Platform oVirt Lab2

In a “Hyper-V Platform Provider” you must configure at least the following parameters: Service Name, Microsoft Hyper-V IP server (“Host” field), user name and password with administration rights on the Microsoft Hyper-V.

We can also indicate the “Timeout” in the connection with Microsoft Hyper-V and specify a range of MAC addresses for creating the virtual desktops.

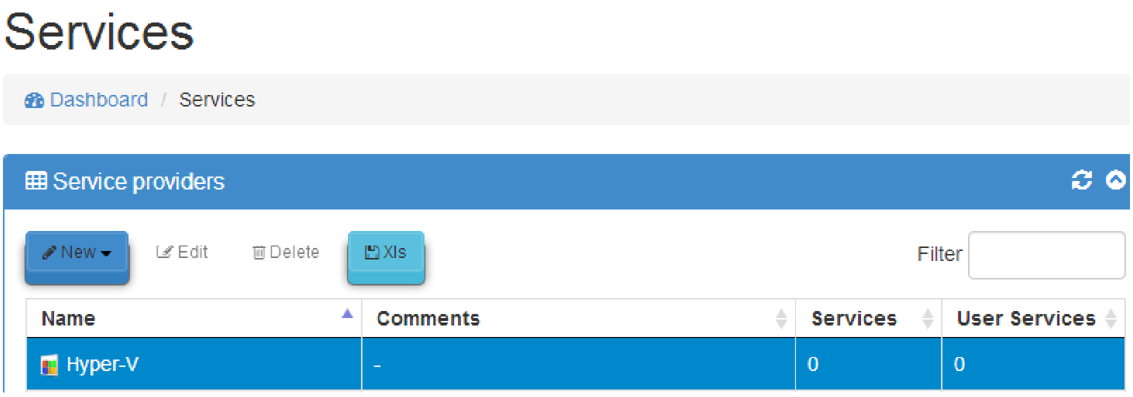
We will check that the connection has been correctly made by clicking on the "Test" button.



New services provider of type **HyperV Platform Provider (experimental)**

Name	Hyper-V
Comments	Comments for this element
Host	192.168.11.80
Port	5985
Username	administrator
Password	••••••••••••
Timeout	64
Macs range	00:15:5D:10:00:00-00:15:5D:FF:FF:FF

When saving this configuration, we already have a valid "Service Providers" to start creating base services in the Microsoft Hyper-V platform. We can register all "Hyper-V Platform Provider" Service Providers we need in the UDS platform.



Services

Dashboard / Services

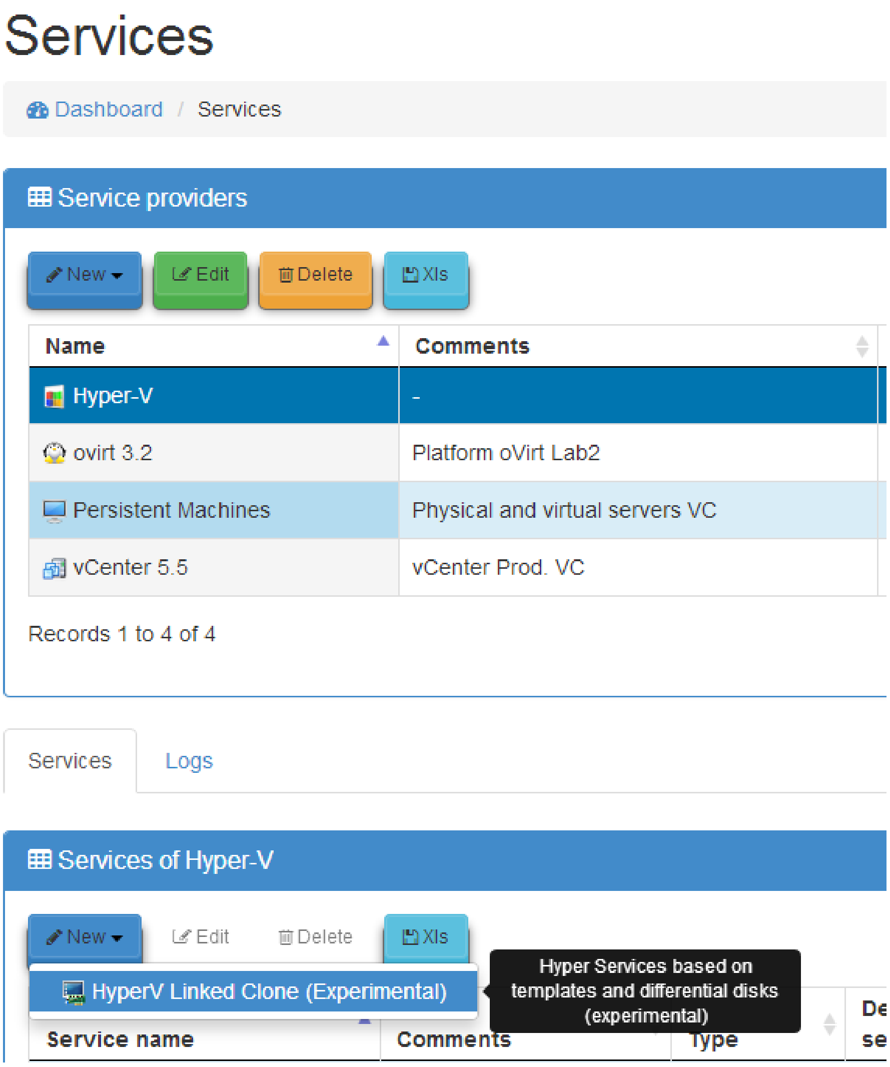
Service providers

Name	Comments	Services	User Services
Hyper-V	-	0	0

4.1.4.2 Configuring service based on “Hyper-V Linked Clone”

Once the Microsoft Hyper-V platform where the desktops will be created has been configured, you must create base services based on "Hyper-V Linked Clone".

Select the Service Providers where we are going to create a "Hyper-V Linked Clone" and click "New".



Services

Dashboard / Services

Service providers

Name	Comments
Hyper-V	-
ovirt 3.2	Platform oVirt Lab2
Persistent Machines	Physical and virtual servers VC
vCenter 5.5	vCenter Prod. VC

Records 1 to 4 of 4

Services of Hyper-V

Hyper Services based on templates and differential disks (experimental)

Service name	Comments	Type	De se

Type a descriptive name for the template and configure the service parameters:

Base Machine: Template for deploying the virtual desktops.

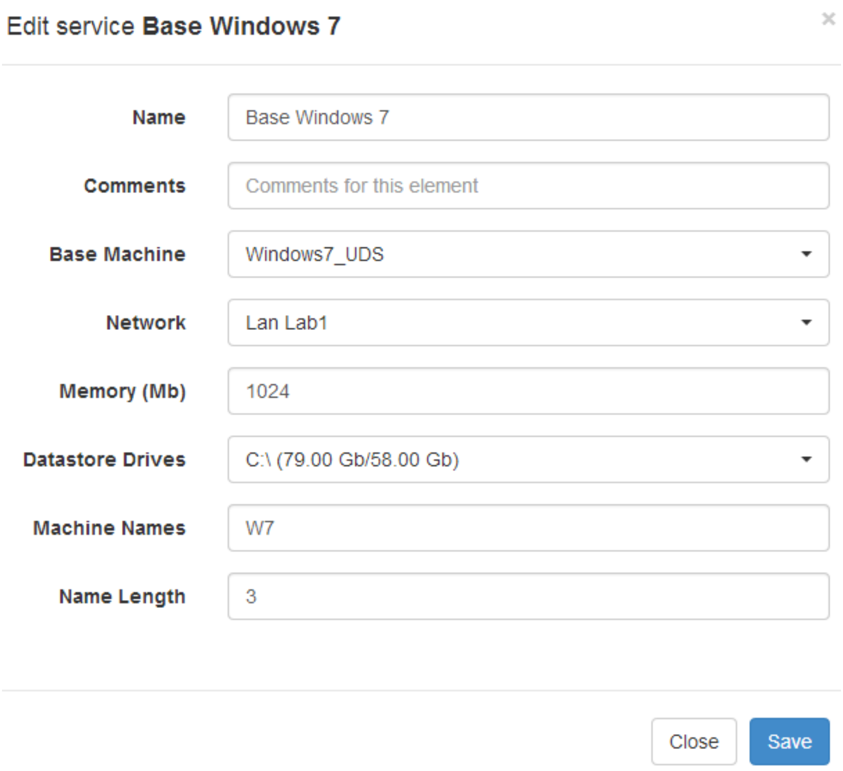
Network: Network to which the desktops will be connected.

Memory: Amount of memory to be assigned to the Linked Clones virtual desktops.

Datastores Drives: Location where the publication of the service and the Linked Clones created will be stored. We can select one, several or all of the datastores clicking the "Ctrl" button. If you select more than one, the system will always locate the new publications and desktops in the Datastore with more free space.

Machine Names: Root name of all of the Linked Clones virtual desktops to be deployed on this service. (ex: Machine Names= W1).

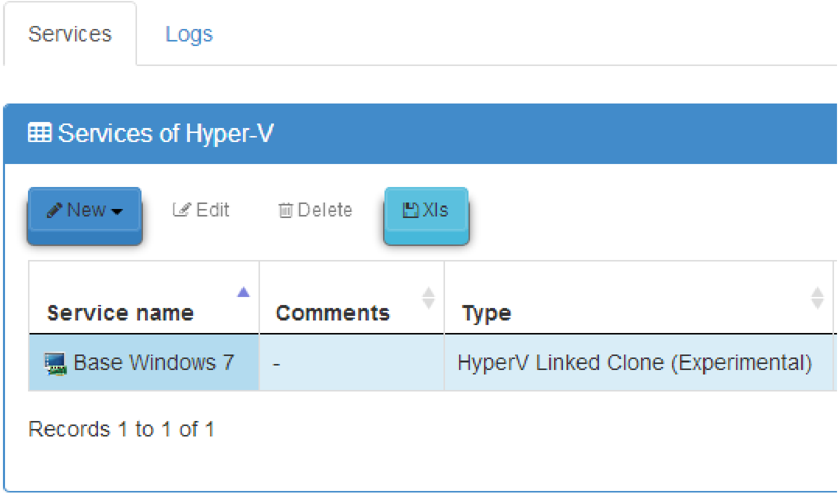
Name Length: Number of digits of the counter attached to the root name of the desktops (ex: Name Length= 3, W7**001**...W7**999**).



Edit service **Base Windows 7**

Name	Base Windows 7
Comments	Comments for this element
Base Machine	Windows7_UDS
Network	Lan Lab1
Memory (Mb)	1024
Datastore Drives	C:\ (79.00 Gb/58.00 Gb)
Machine Names	W7
Name Length	3

When saving this configuration, we already have a valid "Hyper-V Linked Clone" in the Microsoft Hyper-V platform. We can register all "Hyper-V Linked Clone" we need in the UDS platform.

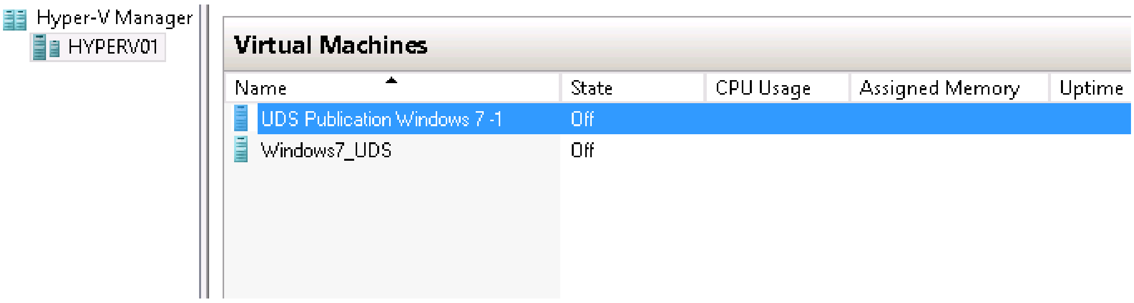


Service name	Comments	Type
Base Windows 7	-	HyperV Linked Clone (Experimental)

Records 1 to 1 of 1

Once the entire UDS environment has been configured and the first Service Pools have been created, we will be able to observe how the virtual desktops based on Microsoft Hyper-V are deployed on the Microsoft Hyper-V server.

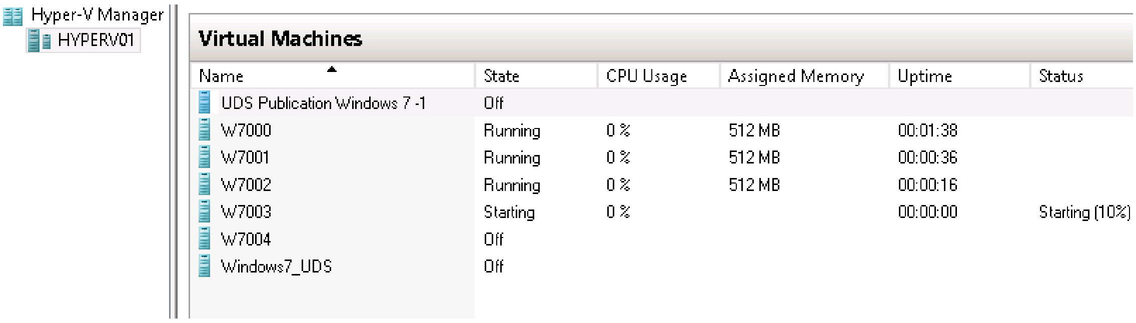
The first task that the Microsoft Hyper-V server will perform will be to create a base machine (this machine will be created each time we make a publication of a service) which will be a clone of the template selected when registering the service, with a hard drive size and characteristics equal to those of said template.



Name	State	CPU Usage	Assigned Memory	Uptime
UDS Publication Windows 7 -1	Off			
Windows7_UDS	Off			

Once the process of creating the base machine has been completed (the UDS system calls it: “UDS Publication name_service –number_publication”), the creation of virtual desktops in the Microsoft Hyper-V automatically begins (the UDS system calls it: “UDS service Machine_Name+Name_Length”.

The hard drive space taken up by the virtual desktops ("Linked Clones") will be exclusively that which is taken up by the changes made by the users on the machines after their initial connection.



Name	State	CPU Usage	Assigned Memory	Uptime	Status
UDS Publication Windows 7 -1	Off				
W7000	Running	0 %	512 MB	00:01:38	
W7001	Running	0 %	512 MB	00:00:36	
W7002	Running	0 %	512 MB	00:00:16	
W7003	Starting	0 %		00:00:00	Starting (10%)
W7004	Off				
Windows7_UDS	Off				

4.1.5 Connection to persistent hardware

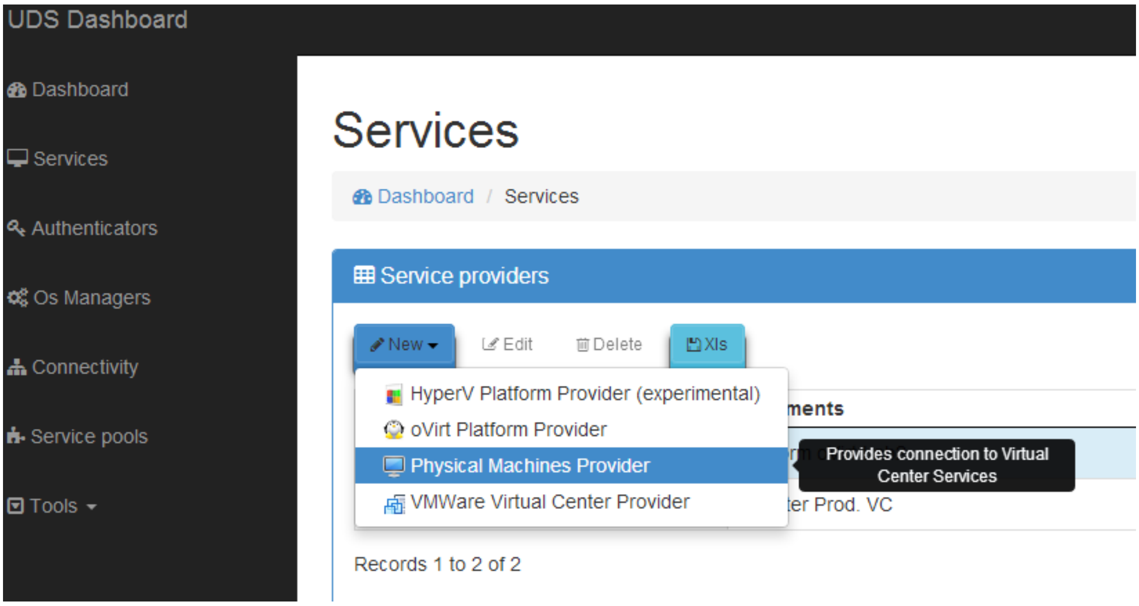
Access persistent hardware by assigning fixed-user IP addresses.

Assigning IP addresses and usernames will be done by order of access, that is, the first user that accesses this service will be assigned the first IP address on the list.

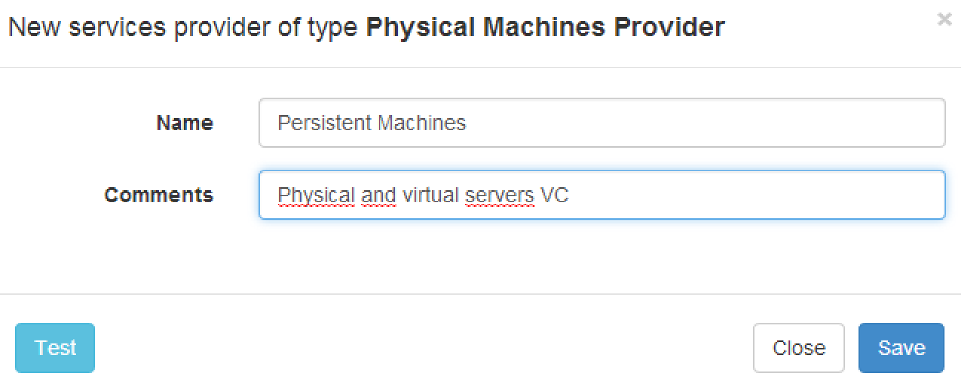
In order to connect to the machine to which the assigned IP address to a user belongs, the machine must have previously been switched on, the Terminal Services for Windows machines must be enabled and the NX software for Linux machines must be installed.

4.1.5.1 Registering service provider "Physical Machine Providers"

Enter "Services", click "New" and select "Physical Machines Provider".



Choose a name for the "Physical Machine Provider".

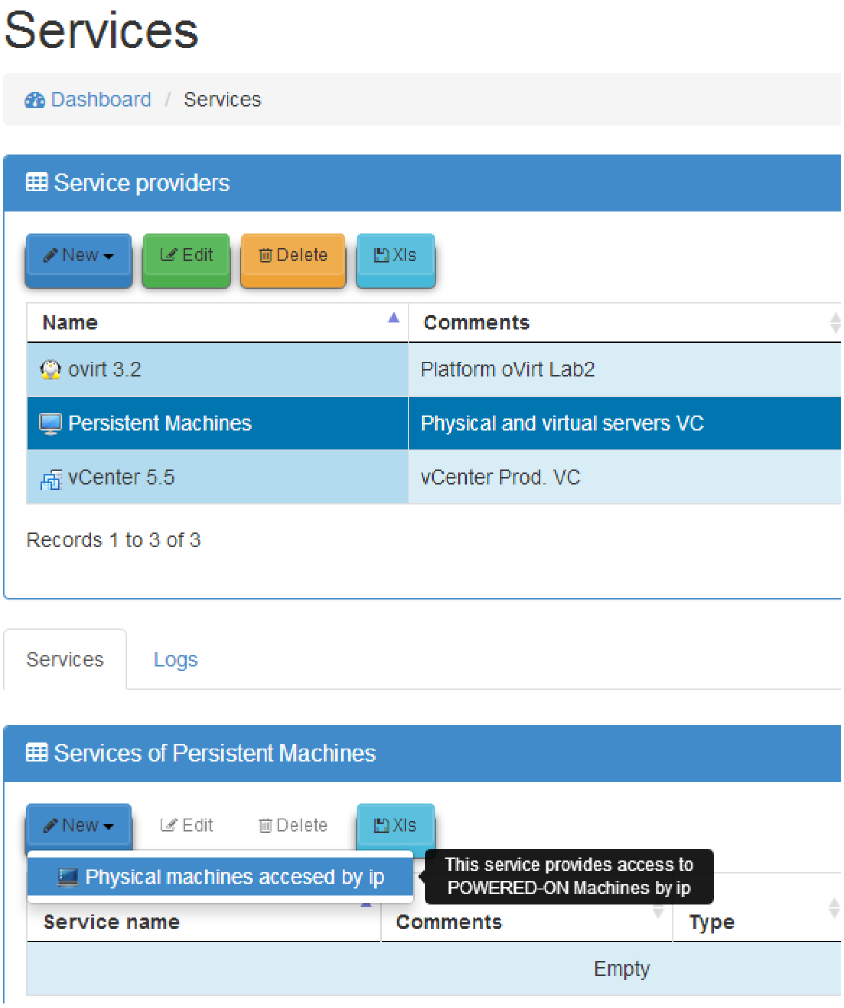

New services provider of type **Physical Machines Provider**

Name	Persistent Machines
Comments	Physical and virtual servers VC

4.1.5.2 Configuring service based on “Physical Machines Providers”

Once the Service Provider for persistent equipment has been created, you must register Base Service based on “Physical machines accessed by IP”.

Select the "Service Providers" where we are going to create a "Physical machines accessed by IP" and click "New".



Services

Dashboard / Services

Service providers

Name	Comments
ovirt 3.2	Platform oVirt Lab2
Persistent Machines	Physical and virtual servers VC
vCenter 5.5	vCenter Prod. VC

Records 1 to 3 of 3

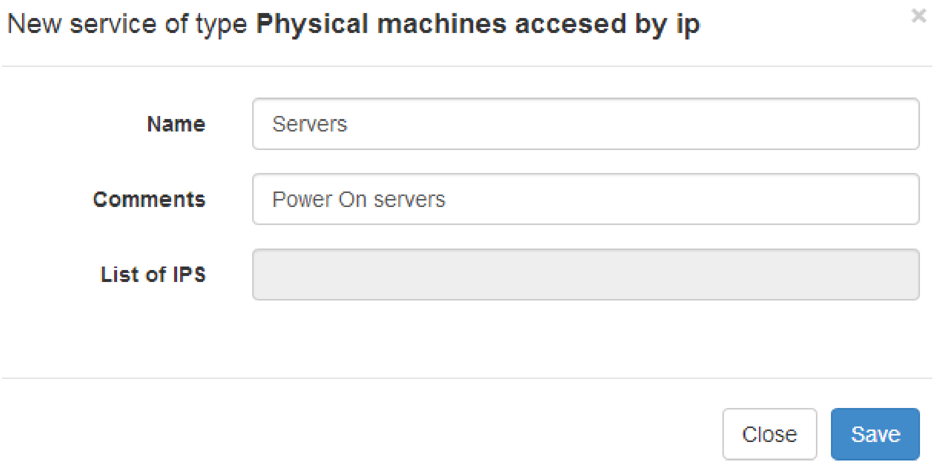
Services of Persistent Machines

This service provides access to POWERED-ON Machines by ip

Service name	Comments	Type
	Empty	

Choose a name for the service and enter the IP addresses to which they will provide access.

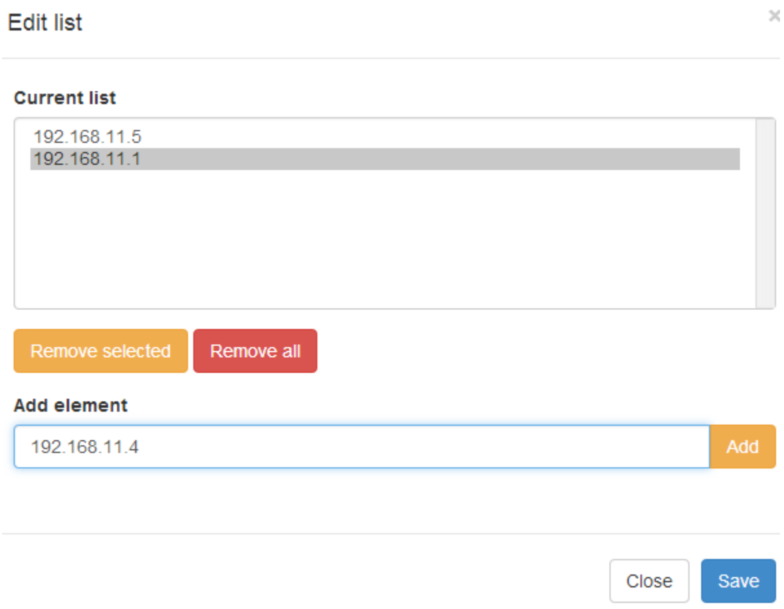
Click "List of IPS" to add IPs addresses:



New service of type **Physical machines accesed by ip**

Name	Servers
Comments	Power On servers
List of IPS	

Enter the IP addresses of the machines to which it will have access and close the editor.



Edit list

Current list

192.168.11.5
192.168.11.1

Add element

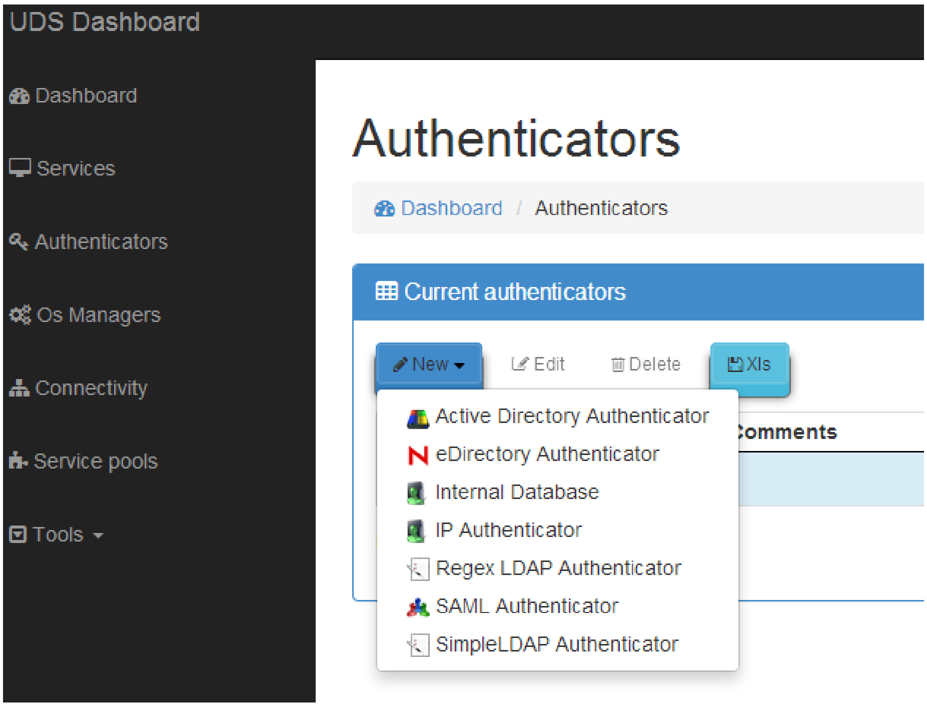
192.168.11.4

4.2 Configuring Authenticators

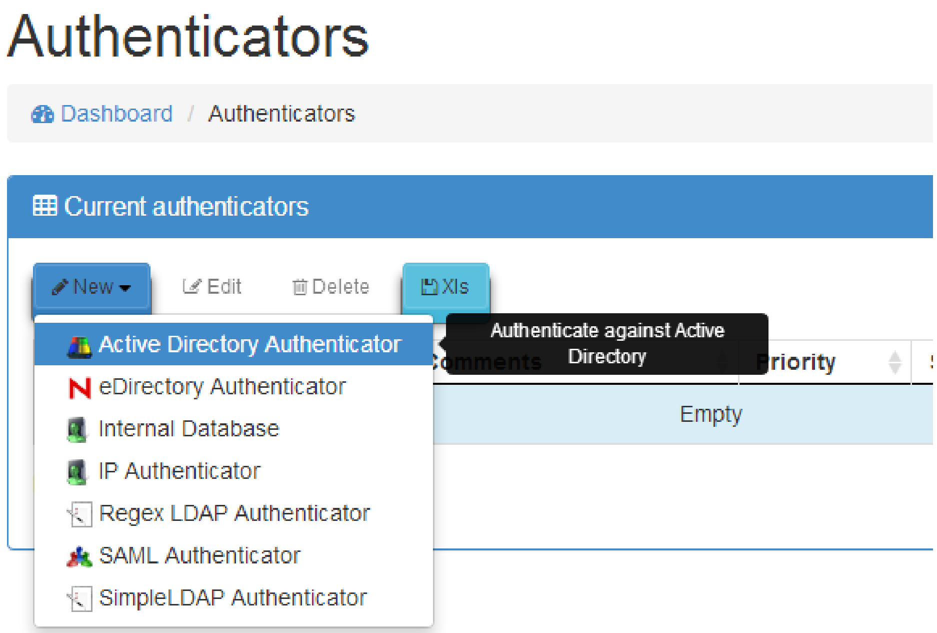
An Authenticator is a basic component within a desktop platform since it allows the users and user groups to whom you have granted sign in credentials to connect to the different virtual desktops.

An Authenticator is not needed to create a Service Pools. But if the Service Pool hasn't at least one authenticator assigned, there will be no users able to connect to UDS platform virtual desktops.

You can choose between different types of authenticators.



4.2.1 Active Directory Authenticator



In an Active Directory Authenticator, we configure the authenticator name, the domain controller IP ("Host" field), a username ("Ldap User" field) and password with reading rights on the Active Directory.

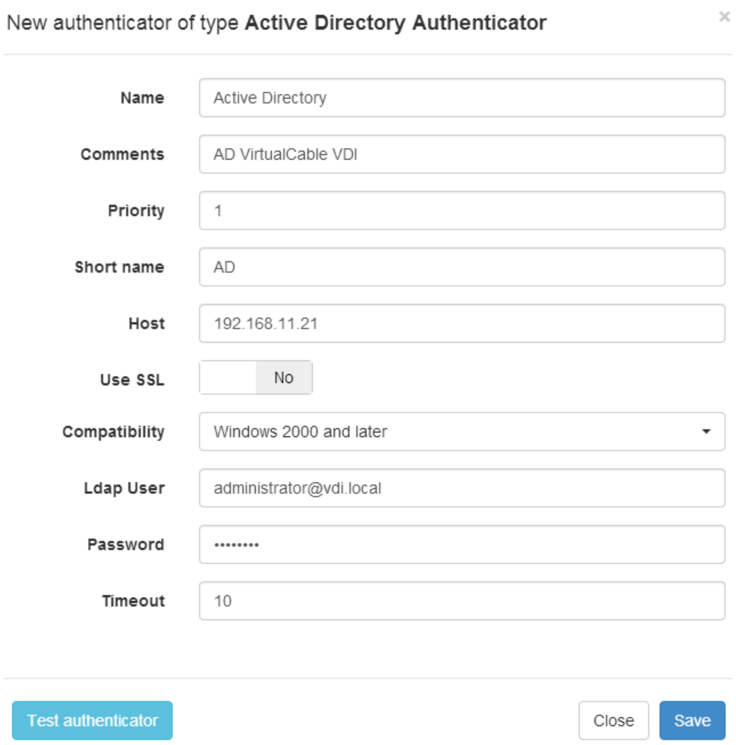
The username ("Ldap User" field) must be typed in with the format *user@domain*.

We can also indicate: the priority of this authenticator, the lower that priority is, the higher it will appear on the list of authenticators available in the user access window (this field admits negative values). We can also choose if we want to use an SSL connection and the Timeout in the connection with the Active Directory.

If the "Small Name" field is chosen, it only allows this "authenticator" to appear on the user login screen; to do this, we have to access the login screen with the following format: address_server_broker/Small_Name

For example: *https://BrokerVC/AD*

By clicking on the “Test” button, we can check to see whether the connection has been made correctly.

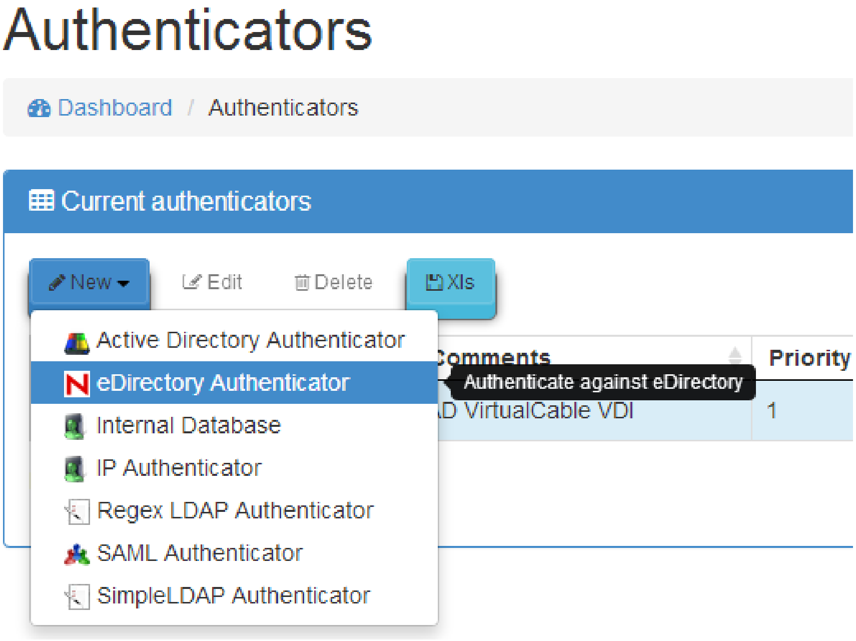


New authenticator of type **Active Directory Authenticator**

Name	Active Directory
Comments	AD VirtualCable VDI
Priority	1
Short name	AD
Host	192.168.11.21
Use SSL	No
Compatibility	Windows 2000 and later
Ldap User	administrator@vdi.local
Password	••••••••
Timeout	10

4.2.2 eDirectory Authenticator

This authenticator is available to provide Novell network users and user groups access to UDS virtual desktops.



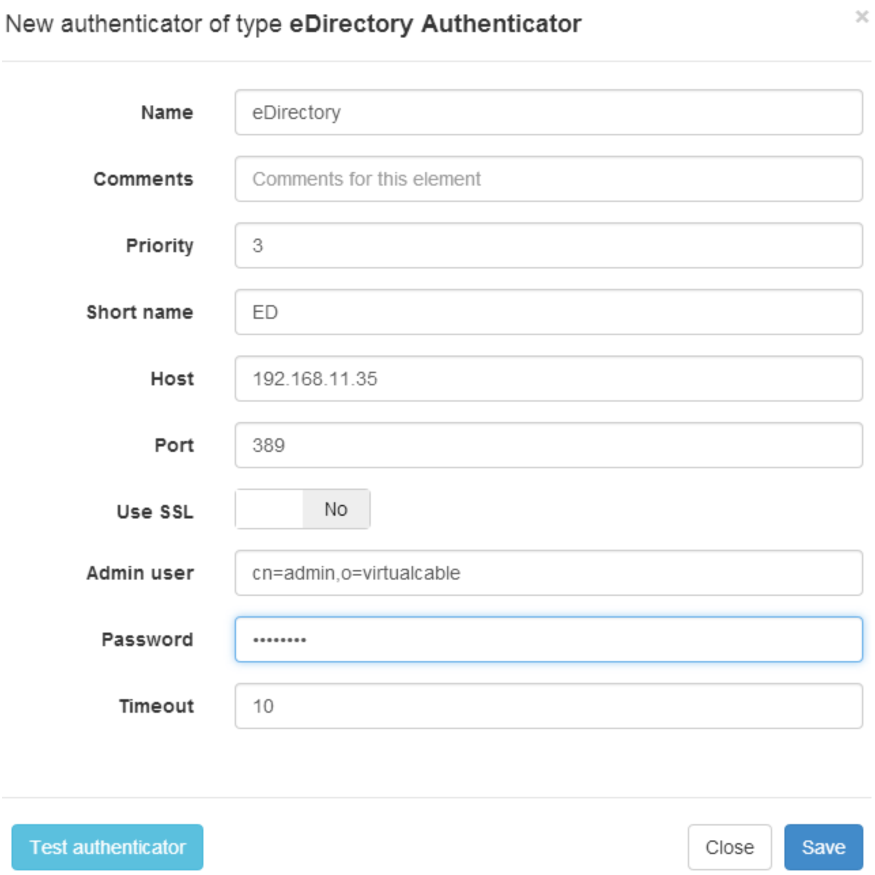
In an eDirectory Authenticator, we configure the authenticator name, the eDirectory IP server (Host field), a username (Admin User field) and password with reading rights on the eDirectory.

We can also indicate: the priority of this authenticator, the lower that priority is, the higher it will appear on the list of authenticators available in the user access window (this field admits negative values). We can also choose if we want to use an SSL connection and the Timeout in the connection with the Active Directory.

If the "Small Name" field is chosen, it only allows this "authenticator" to appear on the user login screen; to do this, we have to access the login screen with the following format: address_server_broker/Small_Name

For example: *https://BrokerVC/ED*

By clicking on the "Test" button, we can check to see whether the connection has been made correctly.

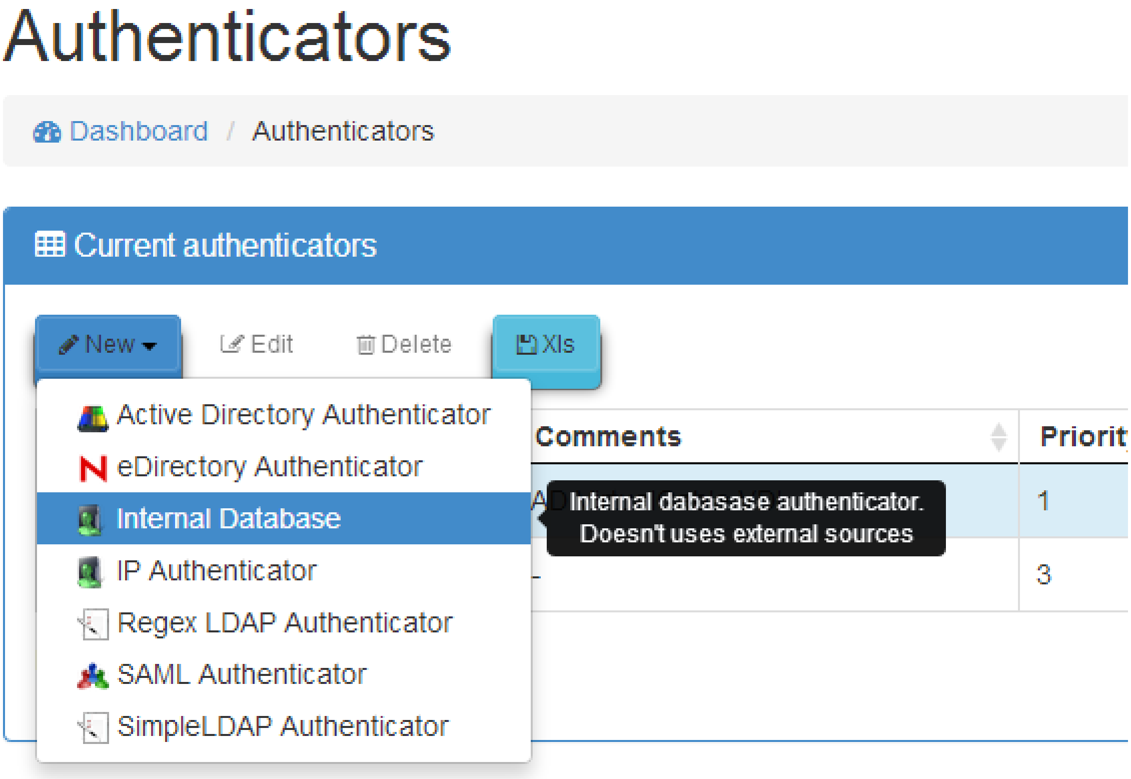


New authenticator of type **eDirectory Authenticator**

Name	eDirectory
Comments	Comments for this element
Priority	3
Short name	ED
Host	192.168.11.35
Port	389
Use SSL	No
Admin user	cn=admin,o=virtualcable
Password	••••••••
Timeout	10

4.2.3 Internal Database

In environments where no external authenticator is available, it is possible to use the internal authenticator. This authenticator is included in the UDS broker and permits you to manually create users and user groups so that they can subsequently access the different Service Pools provided by UDS.



In an Internal Database, we configure the authenticator name.

We can also indicate the priority of this authenticator. The lower that priority is, the higher it will appear on the list of authenticators available in the user access window (this field admits negative values).

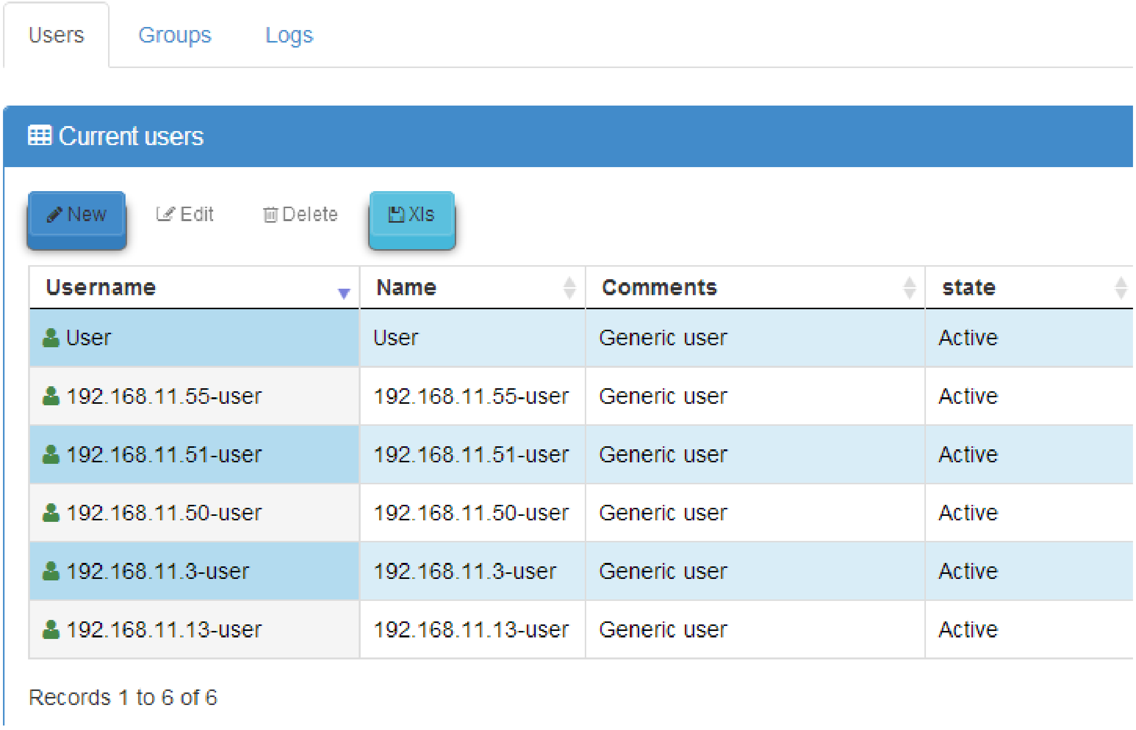
If the “Small Name” field is chosen, it only allows this “authenticator” to appear on the user login screen; to do this, we have to access the login screen with the following format: address_server_broker/Small_Name

For example: https://brokerVC/IN

In this section, we see two options:

"Different user for each host": This option allows connections to virtual desktops by using a single connection user ID. These types of connections are made by creating multiple users in the internal database with the following username structure:

Hostname terminal ID + Connection User ID



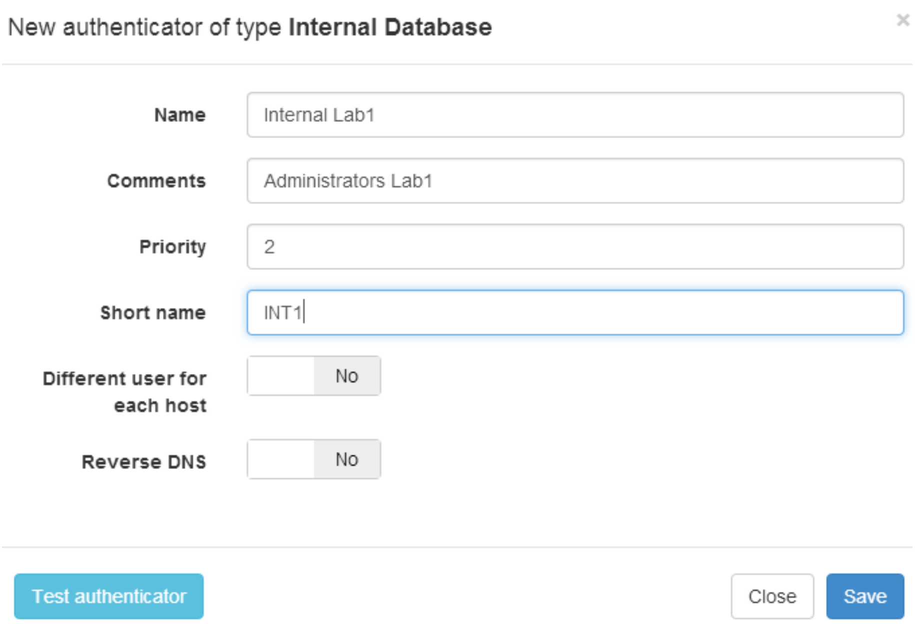
Username	Name	Comments	state
User	User	Generic user	Active
192.168.11.55-user	192.168.11.55-user	Generic user	Active
192.168.11.51-user	192.168.11.51-user	Generic user	Active
192.168.11.50-user	192.168.11.50-user	Generic user	Active
192.168.11.3-user	192.168.11.3-user	Generic user	Active
192.168.11.13-user	192.168.11.13-user	Generic user	Active

Records 1 to 6 of 6

"Reverse DNS": The behavior is exactly the same as in the previous option, but the username structure would be:

Hostname terminal ID + User ID

In order to be able to use this option, you must have the reverse DNS resolution in the connection terminal IDs. In the event that this does not exist, the username structure would continue using the IP address connection terminal ID.



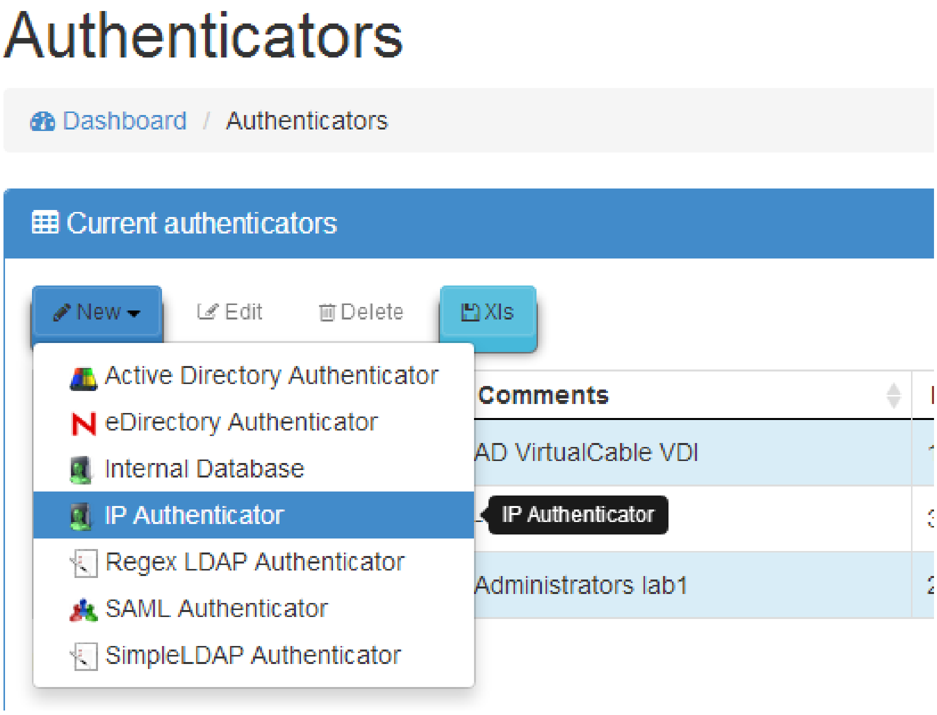
New authenticator of type **Internal Database**

Name	Internal Lab1
Comments	Administrators Lab1
Priority	2
Short name	INT1
Different user for each host	No
Reverse DNS	No

4.2.4 IP Authenticator

It is possible to assign virtual desktops to connecting devices via the IP identifier.

(Ex: Thin Clients in kiosk mode, Call Center environments, proprietary applications, etc...)



Authenticators

Dashboard / Authenticators

Current authenticators

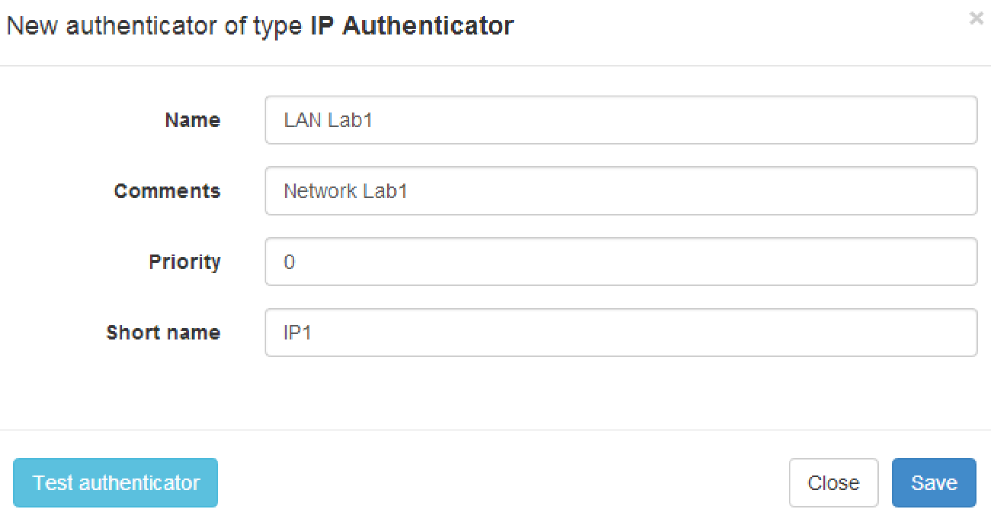
Comments
AD VirtualCable VDI
Administrators lab1

In an IP Authenticator, we configure the authenticator name.

We can also indicate the priority of this authenticator. The lower that priority is, the higher it will appear on the list of authenticators available in the user access window (this field admits negative values).

If the "Small Name" field is chosen, it only allows this "authenticator" to appear on the user login screen; to do this, we have to access the login screen with the following format: address_server_broker/Small_Name

For example: *https://BrokerVC/IP1*

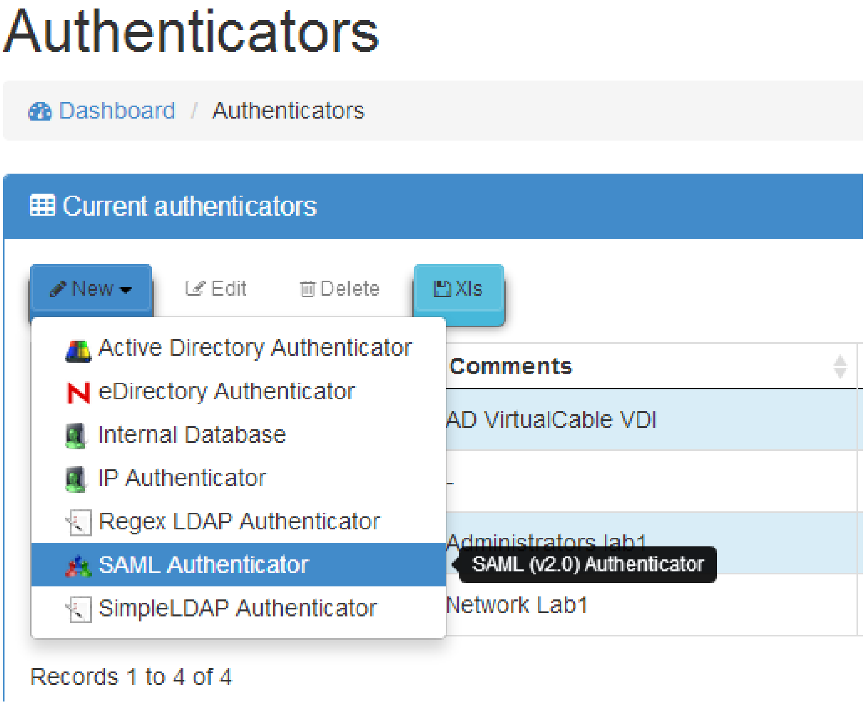


New authenticator of type **IP Authenticator**

Name	LAN Lab1
Comments	Network Lab1
Priority	0
Short name	IP1

4.2.5 SAML Authenticator

SAML is used to exchange authentication and authorization data between security domains, that is, between an identity provider (an assertion producer) and a service provider (an assertion consumer).

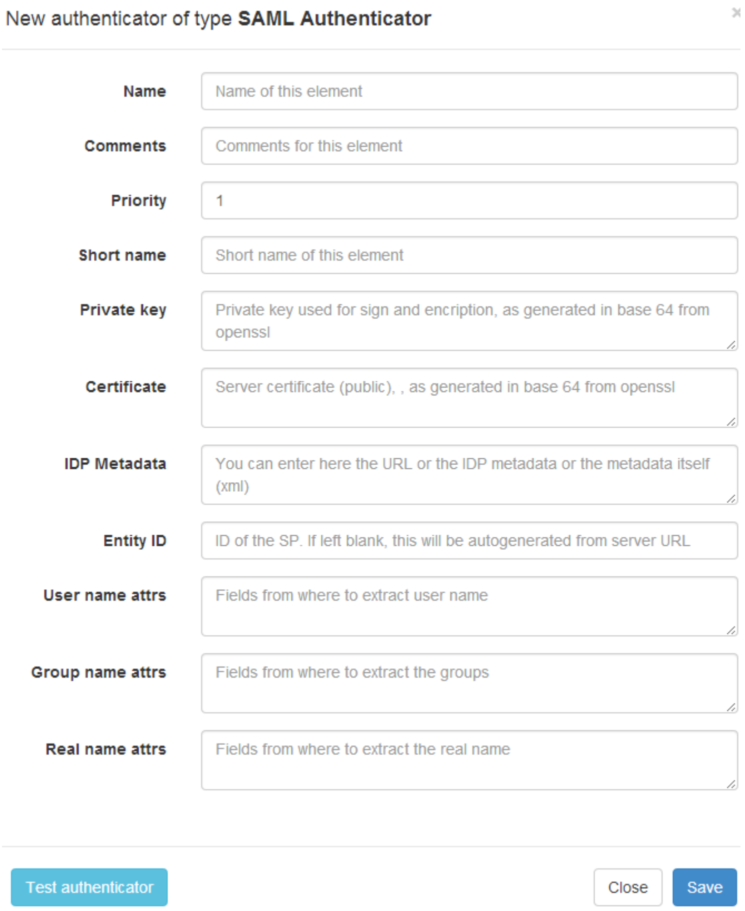


In a SAML Authenticator, we configure the authenticator name and data: Private Key, Certificate IDP Metedata, Entity ID, User name attrs, Group name attrs and Real name attrs.

We can also indicate the priority of this authenticator. The lower that priority is, the higher it will appear on the list of authenticators available in the user access window (this field admits negative values).

If the “Small Name” field is chosen, it only allows this “authenticator” to appear on the user login screen; to do this, we have to access the login screen with the following format: address_server_broker/Small_Name

For example: *https://BrokerVC/SL*

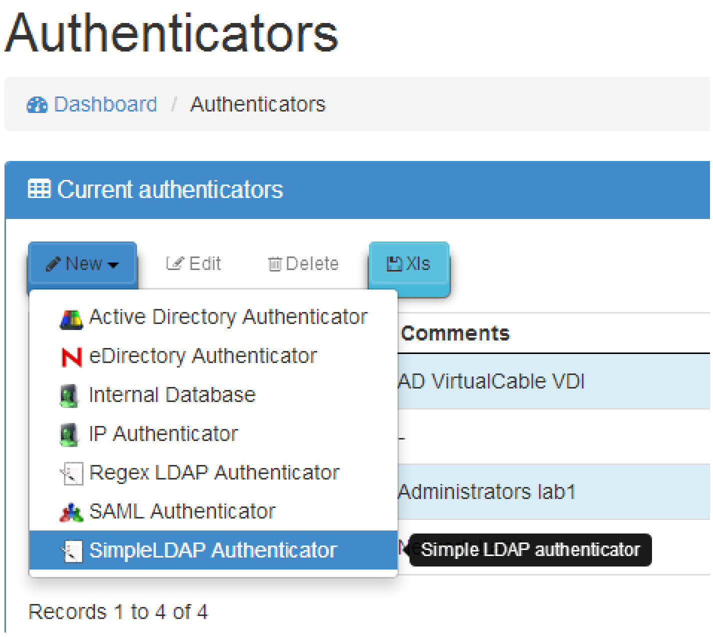


New authenticator of type **SAML Authenticator**

Name	Name of this element
Comments	Comments for this element
Priority	1
Short name	Short name of this element
Private key	Private key used for sign and encription, as generated in base 64 from openssl
Certificate	Server certificate (public), , as generated in base 64 from openssl
IDP Metadata	You can enter here the URL or the IDP metadata or the metadata itself (xml)
Entity ID	ID of the SP. If left blank, this will be autogenerated from server URL
User name attrs	Fields from where to extract user name
Group name attrs	Fields from where to extract the groups
Real name attrs	Fields from where to extract the real name

4.2.6 LDAP Authenticator

This is a generic authenticator available within the UDS platform. By configuring the correct parameters according to each case, we can define practically any authentication service based on LDAP.



Comments
AD VirtualCable VDI
-
Administrators lab1

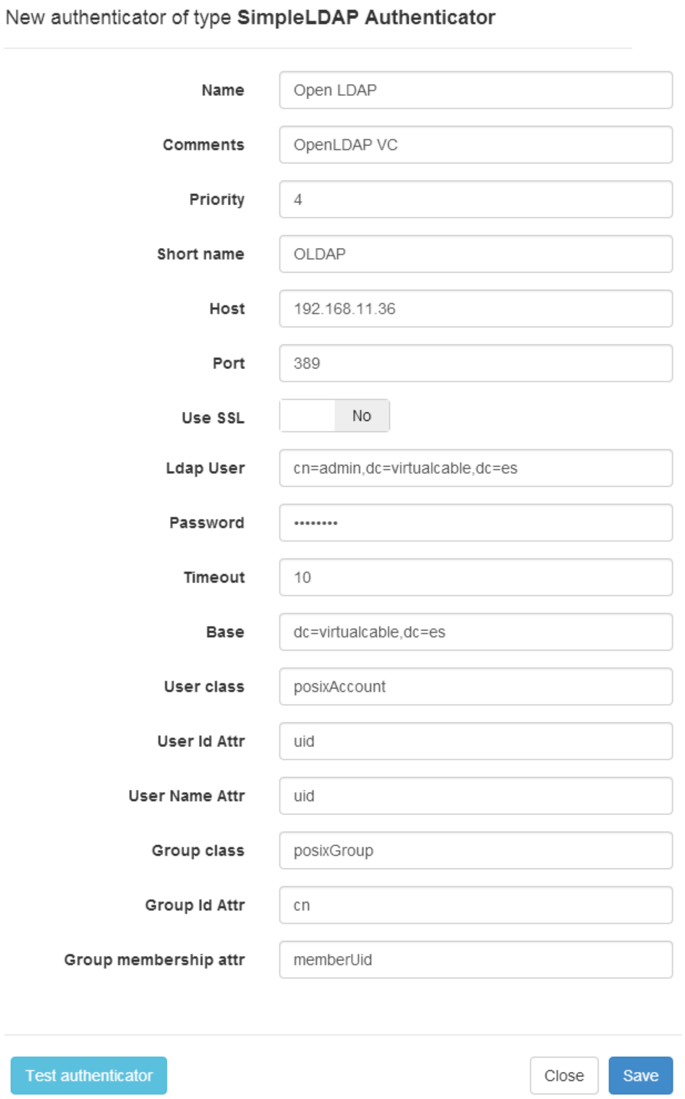
In an LDAP Authenticator (Simple or Regex), we configure the authenticator name, the LDAP server IP (“Host” field), the connection port, a username (“Ldap User” field) and password with reading rights on LDAP, the name of the user and groups search base (“base” field) in the format: *dc=nombre_dominio,dc=extensión_dominio*).

The username (Ldap User field) must be typed in with the format: *cn=user,dc=name_domain,dc=extension_domain*

We can also indicate the priority of this authenticator. The lower that priority is, the higher it will appear on the list of authenticators available in the user access window (this field admits negative values).

If the “Small Name” field is chosen, it only allows this “authenticator” to appear on the user login screen; to do this, we have to access the login screen with the following format: address_server_broker/Small_Name

For example: *https://BrokerVC/OLDAP*



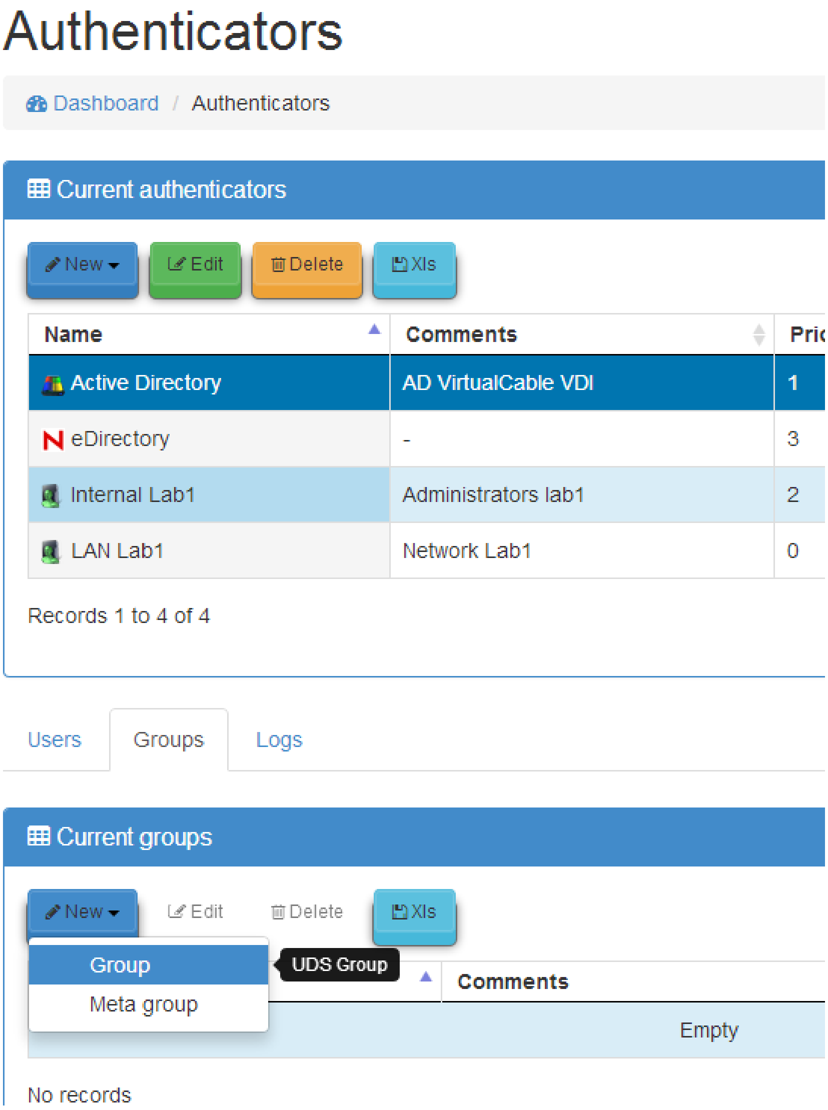
New authenticator of type **SimpleLDAP Authenticator**

Name	Open LDAP
Comments	OpenLDAP VC
Priority	4
Short name	OLDAP
Host	192.168.11.36
Port	389
Use SSL	No
Ldap User	cn=admin,dc=virtualcable,dc=es
Password	••••••••
Timeout	10
Base	dc=virtualcable,dc=es
User class	posixAccount
User Id Attr	uid
User Name Attr	uid
Group class	posixGroup
Group Id Attr	cn
Group membership attr	memberUid

4.3 Configuring users, user groups and user metagroups

Once the authenticator or authenticators have been configured, you must configure the user groups that contain the users to whom access to the virtual desktops is to be granted. It is also possible to create metagroups, which will be used to combine several groups.

To create a group, select the authenticator where we want to create or add the group. In the new window that appears at the bottom of the window, select "Groups" tab and click "new".



Name	Comments	Pri
Active Directory	AD VirtualCable VDI	1
eDirectory	-	3
Internal Lab1	Administrators lab1	2
LAN Lab1	Network Lab1	0

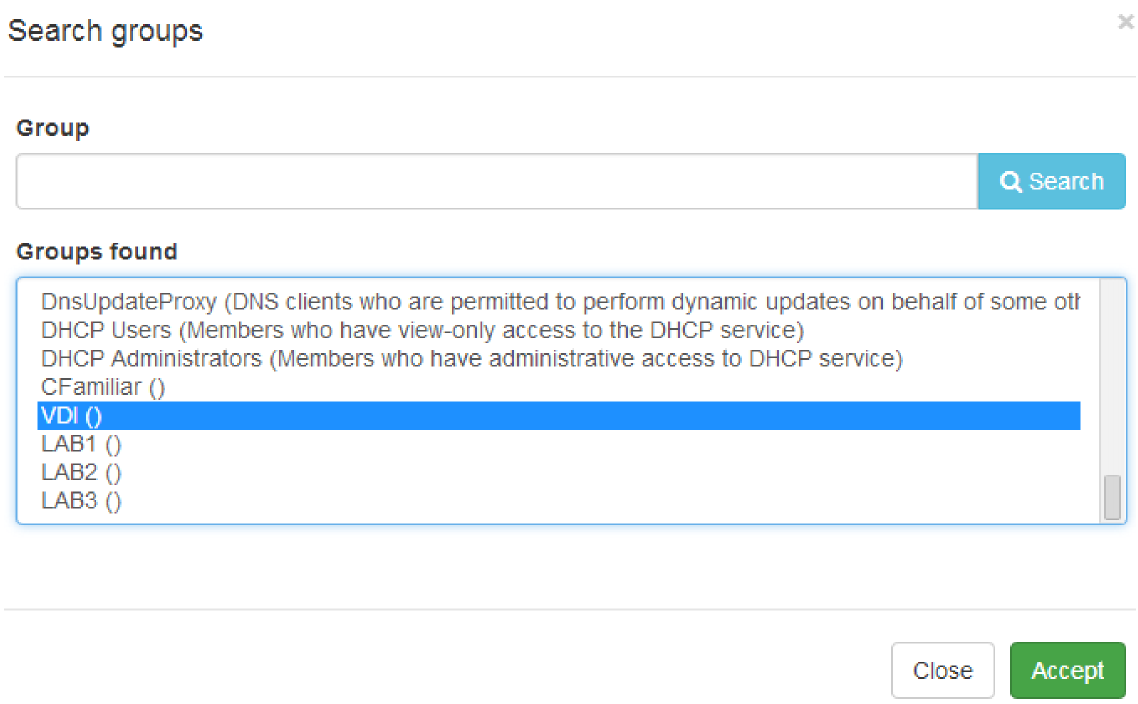
Records 1 to 4 of 4

Empty

No records

Searching for user groups is done automatically in all of the defined authenticators in UDS, with the exception of "Internal" and "by IP" authenticators (Sometimes the search option doesn't work in OpenLdap or eDirectory authenticators. In that case, you may indicate manually the group name), in which the groups are registered without being able to perform a search.

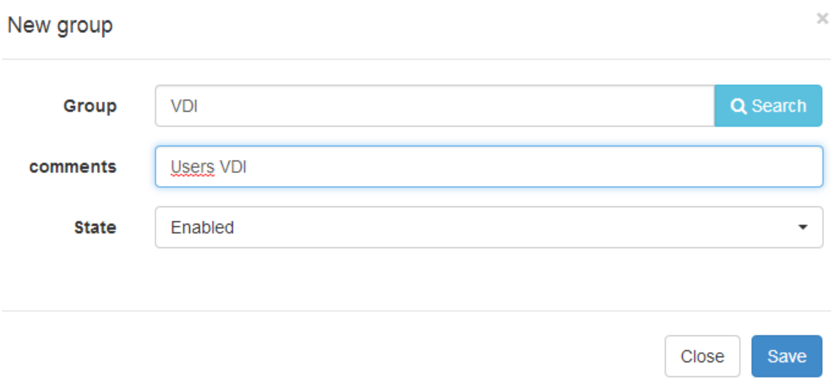
To look for a group, click "Search". We can write down a root name to enclose the search. If we leave this field empty, all the available groups on the authenticator will appear. If we need to add more than one group, we'll have to do it one by one.



DnsUpdateProxy (DNS clients who are permitted to perform dynamic updates on behalf of some otl
DHCP Users (Members who have view-only access to the DHCP service)
DHCP Administrators (Members who have administrative access to DHCP service)
CFamiliar ()
VDI ()
LAB1 ()
LAB2 ()
LAB3 ()

Once the group is selected, click "Accept". If you know the name, you can write it down directly, but it is recommended to check that it appears in the right way in the search option.

The groups, metagroups and users can be temporarily activated or deactivated.

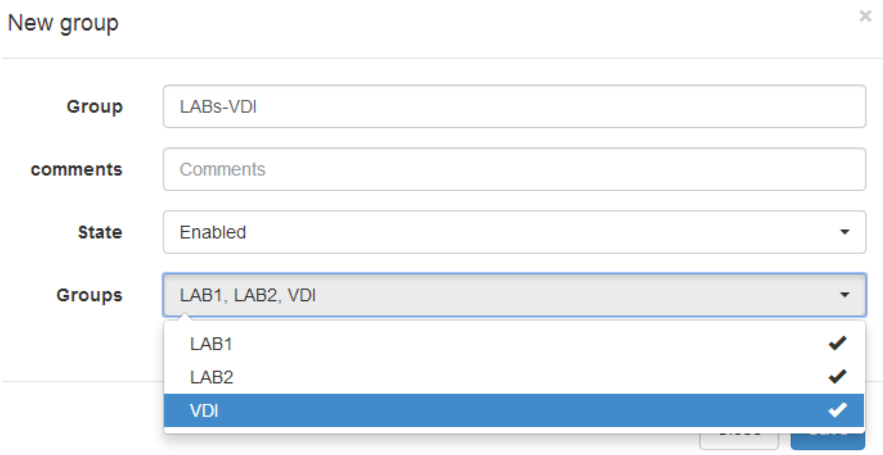


New group

Group	VDI
comments	Users VDI
State	Enabled

To create a metagroup, we select the groups that will form part of the metagroup, we choose a name for the new group and we click "Accept."

A user will belong to this metagroup if he belongs to all the groups which form the metagroup.



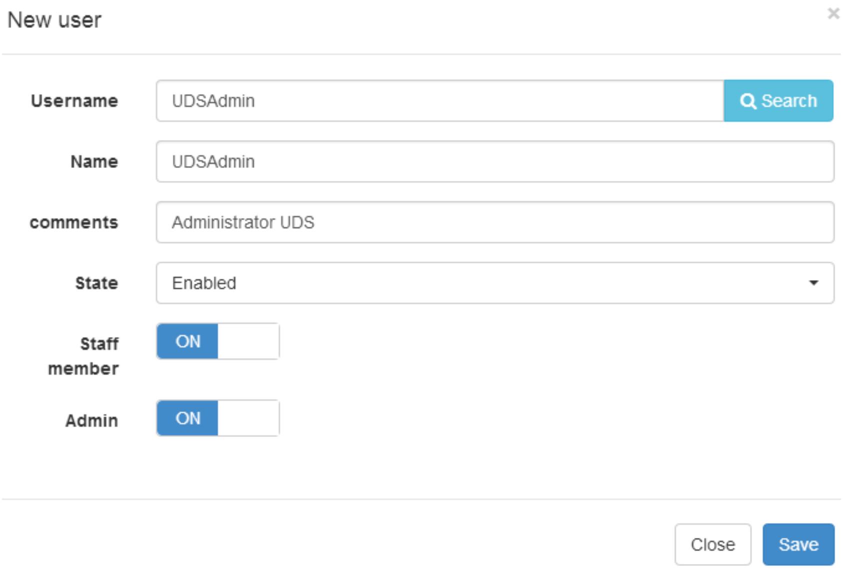
New group

Group	LABs-VDI
comments	Comments
State	Enabled
Groups	LAB1, LAB2, VDI

LAB1 ✓
LAB2 ✓
VDI ✓

The users of the configured groups are automatically added to the system when they connect to the UDS platform for the first time, except in "Internal" or "by IP" authenticators, in which the users will have to be manually registered.

If we need to register new users manually, to assign special permissions, before they connect for the first time and they add themselves automatically, we'll have to select the authenticator and in "Users" tab click "New".



New user

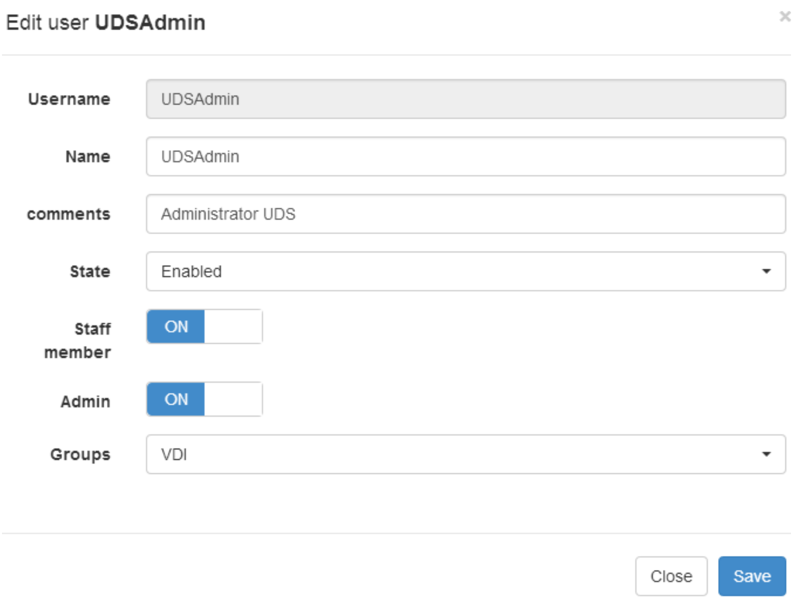
Username	UDSAdmin
Name	UDSAdmin
comments	Administrator UDS
State	Enabled
Staff member	ON
Admin	ON

The additional "Staff" parameter allows access to downloads (UDS actor) and to the UDS administration.

The additional "Admin" parameter allows access to downloads (UDS actor), to the administration and also allows for the modification of advanced UDS configurations (Tab "Tools" - "Configuration"). An "Admin" user has to simultaneously be a "Staff" member.

By clicking "Search" button we can search users created in the authenticator and add them.

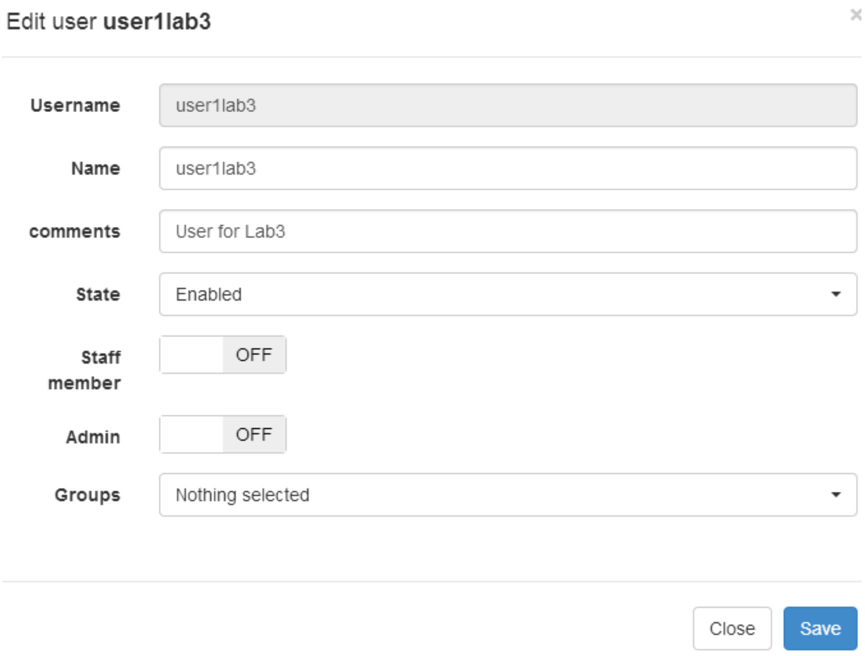
Once the user has been created, click "Edit" to check that the user has been automatically assigned to the group it belongs.



Edit user **UDSAdmin**

Username	UDSAdmin
Name	UDSAdmin
comments	Administrator UDS
State	Enabled
Staff member	ON
Admin	ON
Groups	VDI

If we register a user that belongs to a group which is not registered in the authenticator, it will appear without group and we won't be able to use that user.



Edit user **user1lab3**

Username	user1lab3
Name	user1lab3
comments	User for Lab3
State	Enabled
Staff member	OFF
Admin	OFF
Groups	Nothing selected

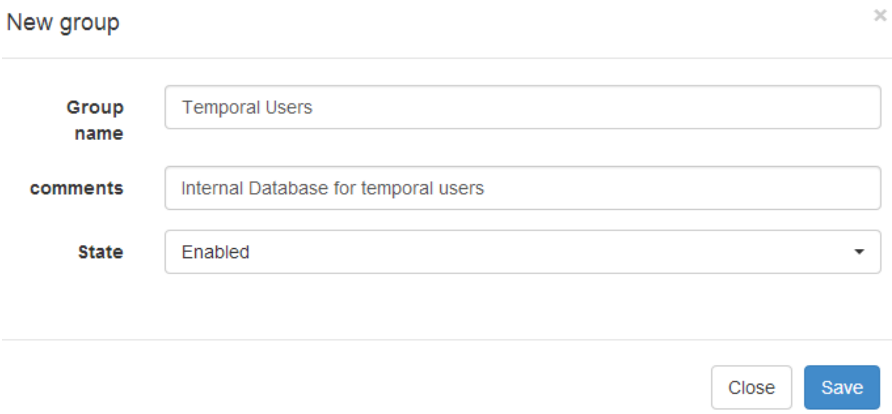
The "Staff member" and "Admin" user options can be modified anytime.

To delete a user, a group or a metagroup, select it and click "Delete" button. If we have registered users in the system that belong to a group and this group is deleted, the users won't have an assigned group and they couldn't be validated in the system.

4.3.1 Creating "Internal Database" groups and users

In an Internal Database authenticator the first thing we have to do is create a group or groups of users.

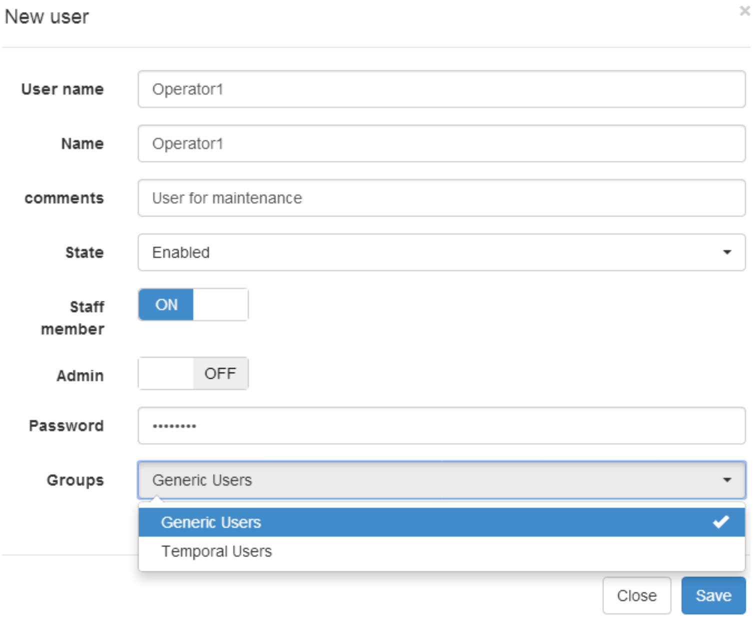
Select the Internal Database authenticator and in "Groups" tab click "New".



New group

Group name	Temporal Users
comments	Internal Database for temporal users
State	Enabled

Once the group or groups of users are created, register the users and assign them to one or several groups. Select the Internal Database authenticator and in “Users” tab click "New".



New user

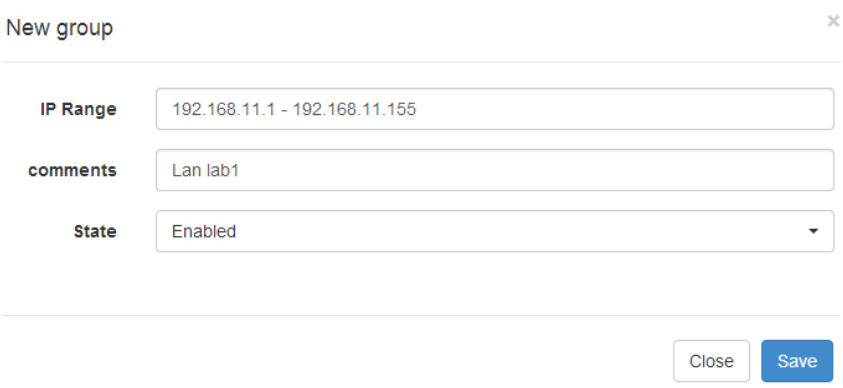
User name	Operator1
Name	Operator1
comments	User for maintenance
State	Enabled
Staff member	ON
Admin	OFF
Password	••••••••
Groups	Generic Users

4.3.2 Creating "IP Authenticator" groups and users

The creation of a group in the “by IP” authenticator is different from the other ones, because in this case it will be a range of IPs addresses which will be registered to allow access to all the hardware within this range. This range of addresses is defined as follows:

IP address start range – IP address end range

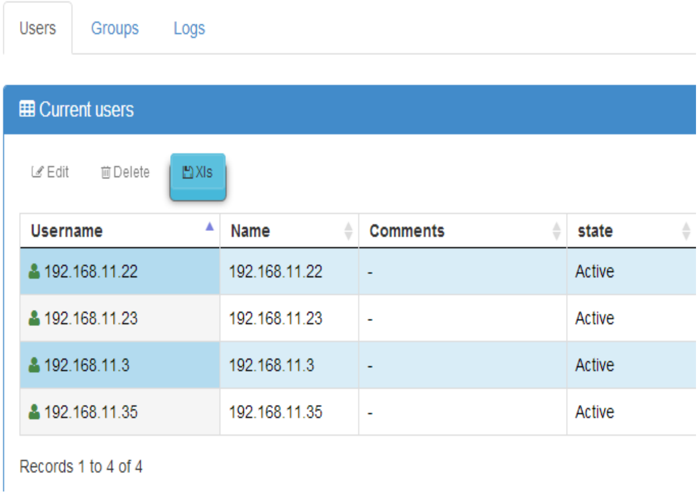
Select “IP Authenticator" authenticator and in "Groups" tab click "New".



New group

IP Range	192.168.11.1 - 192.168.11.155
comments	Lan lab1
State	Enabled

The IP addresses that will be those of the new users, will be automatically added to the "Users" tab the first time they log in.



Username	Name	Comments	state
192.168.11.22	192.168.11.22	-	Active
192.168.11.23	192.168.11.23	-	Active
192.168.11.3	192.168.11.3	-	Active
192.168.11.35	192.168.11.35	-	Active

Records 1 to 4 of 4

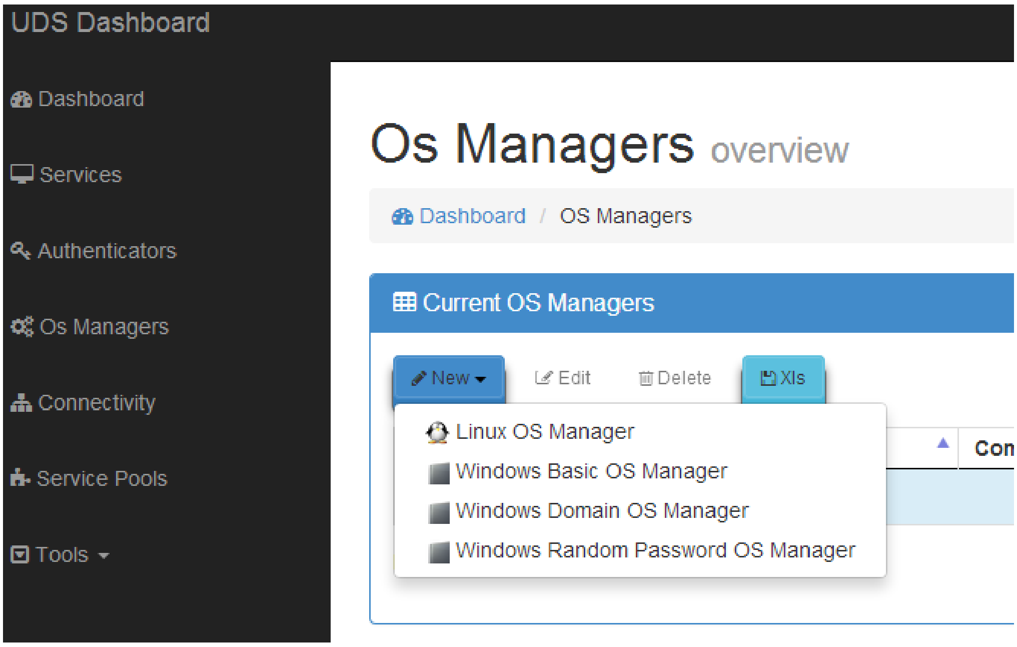
4.4 Configuring "OS Managers"

An OS Manager initiates a previously configured type of service.

The UDS Actor, hosted on the virtual desktop, is responsible for the interaction between the OS and the broker based on the configurations or type of OS Manager chosen.

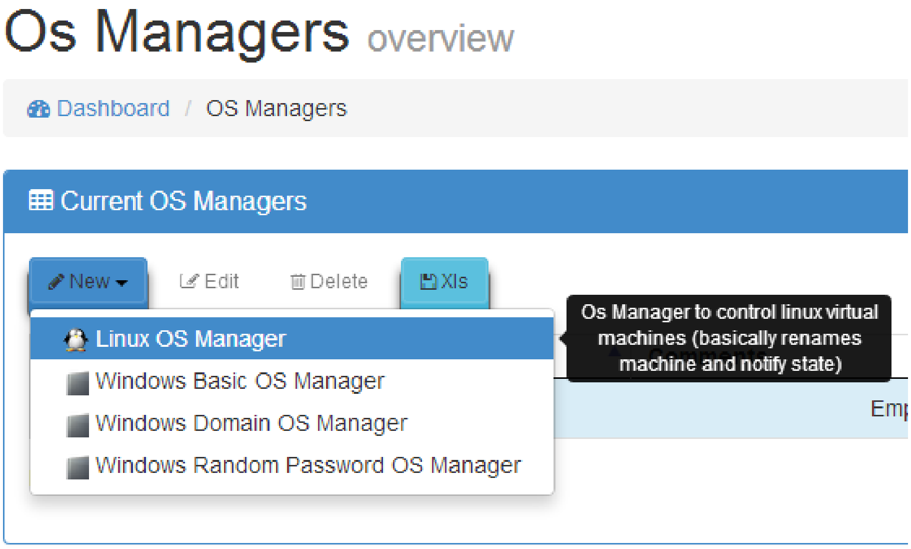
In order to perform VDI deployments via Linked Clones, you will have to select the disconnection behavior of the Linked Clones, within the configuration of each OS Manager.

You can choose different types of "OS Managers".



4.4.1 Linux OS Manager

A "Linux OS Manager" is used for virtual desktops based on Linux systems.

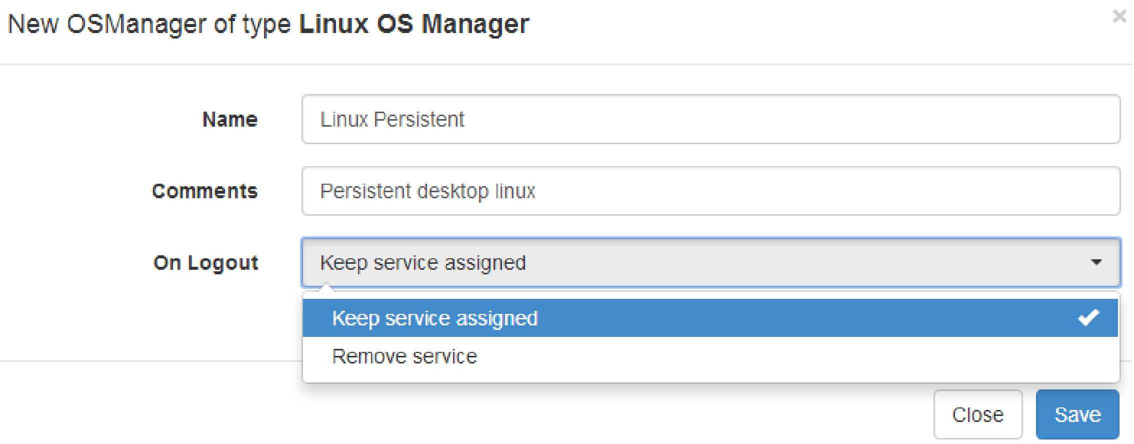


Os Manager to control linux virtual machines (basically renames machine and notify state)

In order to configure Linux OS Manager, enter the name and configure which action the system will perform when a user disconnects:

- Keep service assigned: When a user logs out the desktop won't undergo any change. If this same user requests again a virtual machine to the system, the system will provide the same virtual desktop.

- Remove service: When a user logs out, the system will destroy the desktop. If this same user requests again a virtual machine to the system, the system will provide a new virtual desktop.



New OSManager of type **Linux OS Manager**

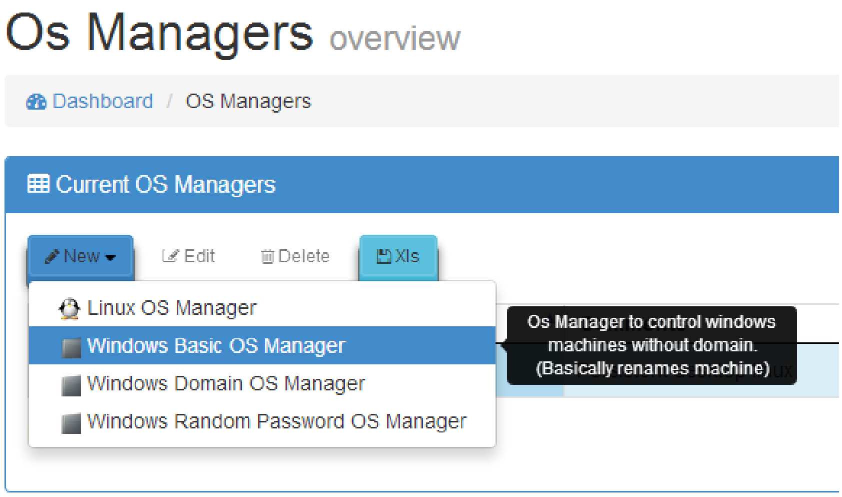
Name: Linux Persistent

Comments: Persistent desktop linux

On Logout: Keep service assigned

4.4.2 Windows Basic OS Manager

A "Windows Basic OS Manager" is used for virtual desktops based on Windows systems which aren't part of a domain.



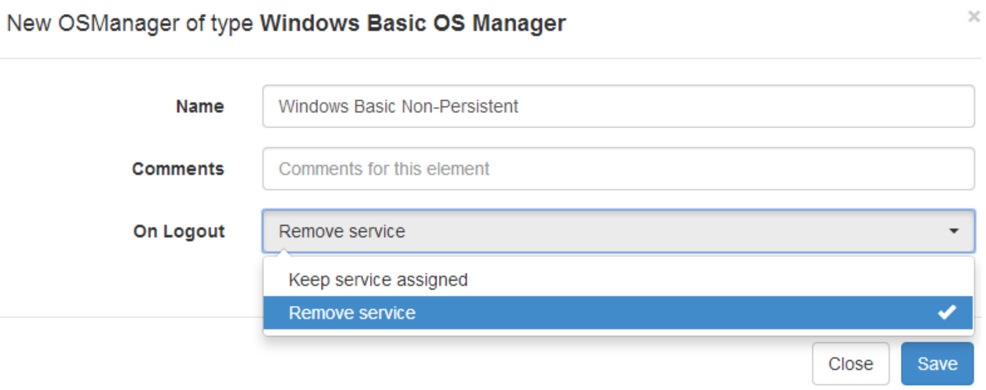
Os Managers overview

Os Manager to control windows machines without domain. (Basically renames machine)

In order to configure a Windows Basic OS Manager, enter the name and configure which action the system will perform when a user disconnects.

- Keep service assigned: When a user logs out the desktop won't undergo any change. If this same user requests again a virtual machine to the system, the system will provide the same virtual desktop.

- Remove service: When a user logs out, the system will destroy the desktop. If this same user requests again a virtual machine to the system, the system will provide a new virtual desktop.



New OSManager of type **Windows Basic OS Manager**

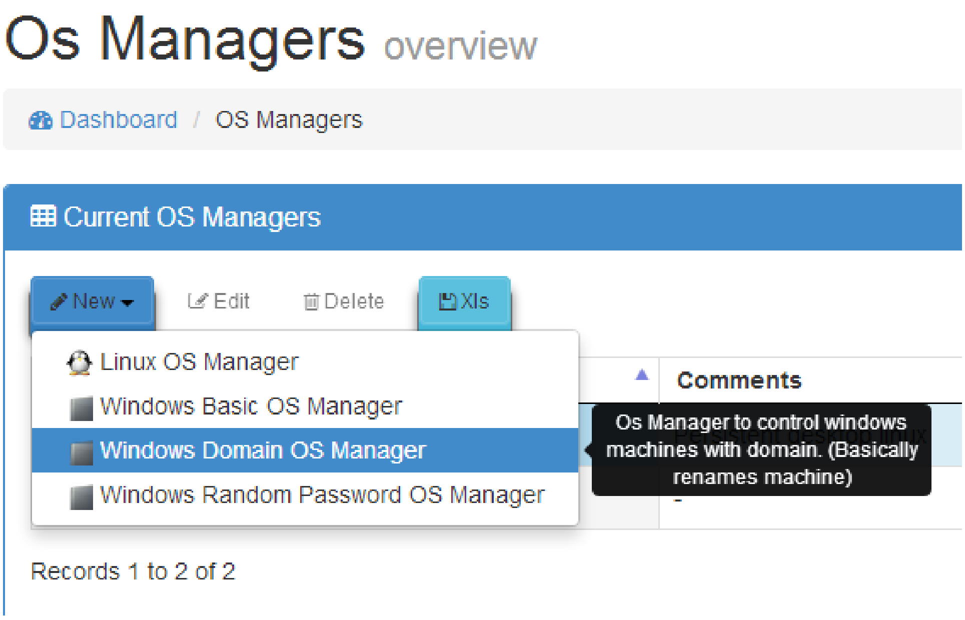
Name: Windows Basic Non-Persistent

Comments: Comments for this element

On Logout: Remove service

4.4.3 Windows Domain OS Manager

A "Windows Domain OS Manager" is used for virtual desktops based on Windows systems which are part of a domain.



Os Managers overview

Dashboard / OS Managers

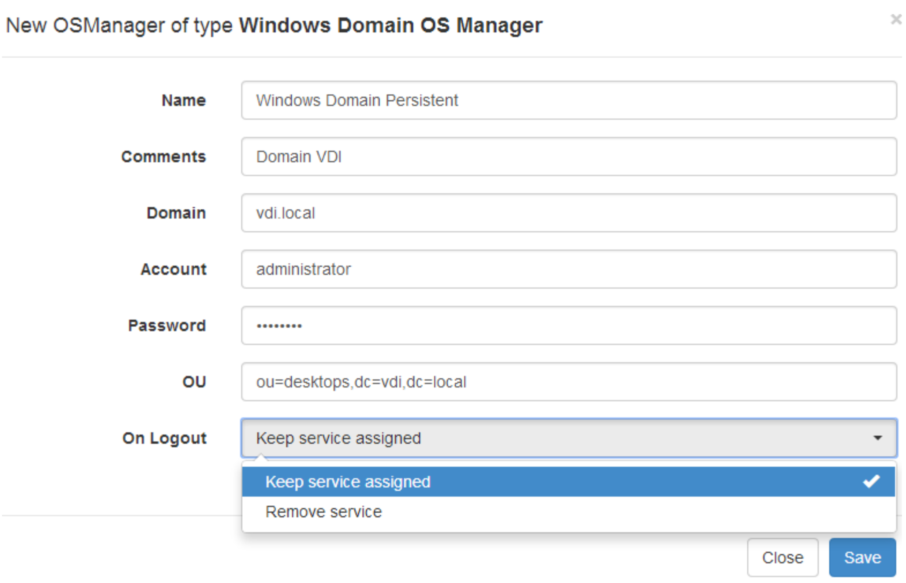
Current OS Managers

Os Manager to control windows machines with domain. (Basically renames machine)

Records 1 to 2 of 2

In order to configure a "Windows Domain OS Manager", enter the following data:

- The name of the OS Manager.
- The domain name to which the virtual desktops deployed with this OS Manager are going to belong.
- User credentials with permission to add machines to the domain.
- Information of the Organizing Unit (OU) where the virtual desktops deployed with this OS Manager are going to be registered (if we don't write anything, the desktops will be located in the branch by default).
- Configure the action the system will perform when a user logs out:
    - Keep service assigned: When a user logs out the desktop won't undergo any change. If this same user requests again a virtual machine to the system, the system will provide the same virtual desktop.
    - Remove service: When a user logs out, the system will destroy the desktop. If this same user requests again a virtual machine to the system, the system will provide a new virtual desktop.

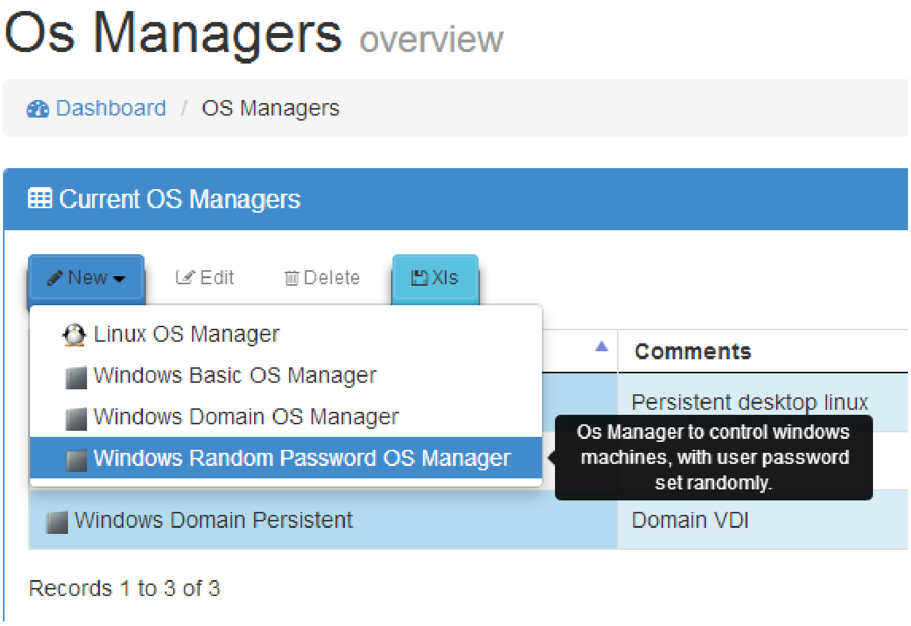


New OSManager of type **Windows Domain OS Manager**

Name	Windows Domain Persistent
Comments	Domain VDI
Domain	vdi.local
Account	administrator
Password	••••••••
OU	ou=desktops,dc=vdi,dc=local
On Logout	Keep service assigned

4.4.4 Windows Random Password OS Manager

A "Windows Random Password OS Manager" is used for virtual desktops based on Windows systems that are not part of a domain and require a higher level of security in the user access.


Os Managers overview

Dashboard / OS Managers

Current OS Managers

Os Manager to control windows machines, with user password set randomly.

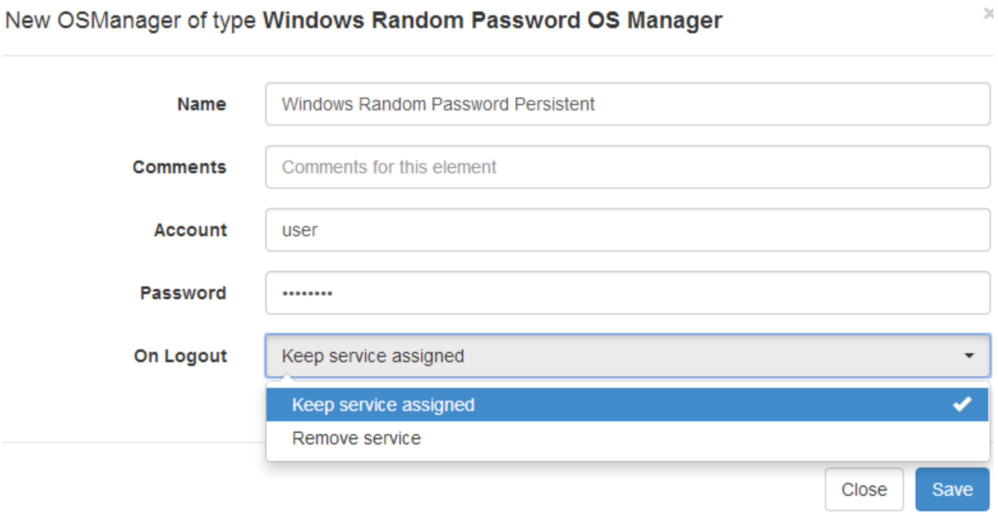
	Comments
	Persistent desktop linux
Windows Domain Persistent	Domain VDI

Records 1 to 3 of 3

Using this assigns a random password, previously defined during configuration, to an existing local user in each new deployed virtual desktop, thus providing a higher level of access security.

To configure a "Windows Random Password OS Manager" enter the following data:

- OS Manager Name.
- Local user defined in the template.
- Password initially established for the local user in the template.
- Configure which action the system will perform when a user disconnects:
  - Keep service assigned: When a user logs out the desktop won't undergo any change. If this same user requests again a virtual machine to the system, the system will provide the same virtual desktop.
  - Remove service: When a user logs out, the system will destroy the desktop. If this same user requests again a virtual machine to the system, the system will provide a new virtual desktop.


New OSManager of type **Windows Random Password OS Manager**

Name: Windows Random Password Persistent

Comments: Comments for this element

Account: user

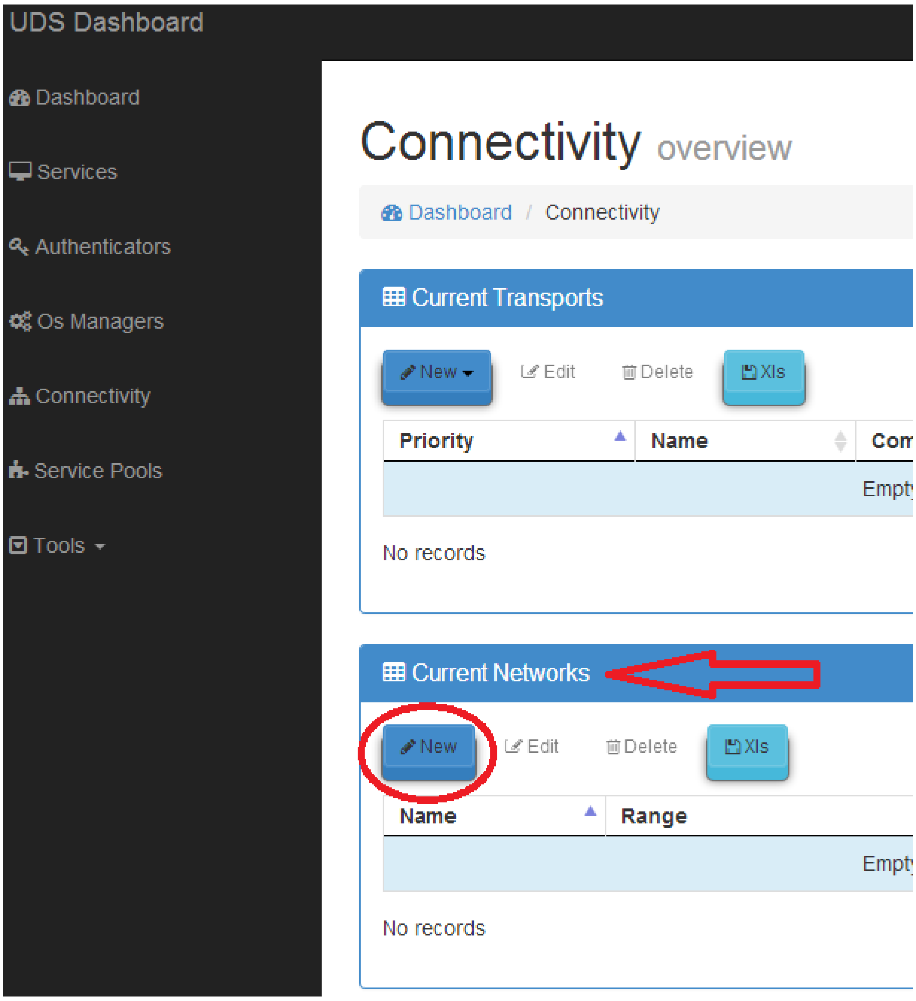
Password: ••••••••

On Logout: Keep service assigned

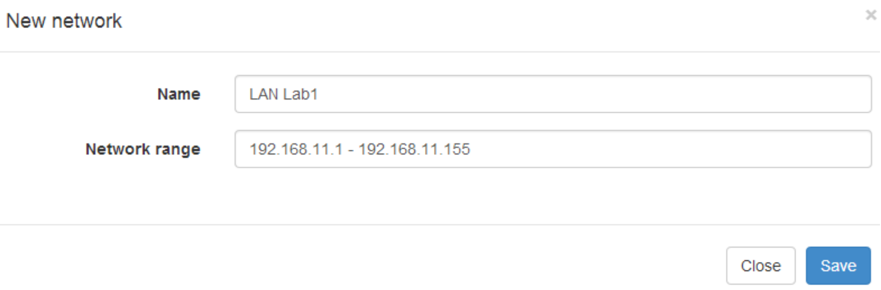
4.5 Configuring “Networks”

UDS allows registering several networks to allow or deny access to virtual desktops. These networks, together with Transports will define what kind of access the users will have to their virtual desktops generated by UDS.

To add a network, go to "Connectivity" section and click "New" in "Current Networks" section.



Write a descriptive name and a network range (most existing formats are supported).



New network ×

Name: LAN Lab1

Network range: 192.168.11.1 - 192.168.11.155

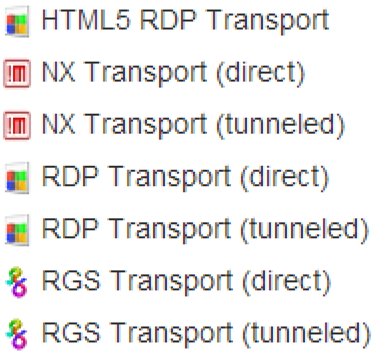
If no network is registered, access to the virtual desktops will be allowed from any network.

4.6 Configuring "Transports"

In order to connect to the virtual desktops, you must create Transports. These are small applications that will be run on the client and which will be responsible for providing access to the implemented service.

Depending on what type of virtual desktop is configured, the location and way of connection to our virtual desktops, we must create different types of transports.

The following Transports are available nowadays:



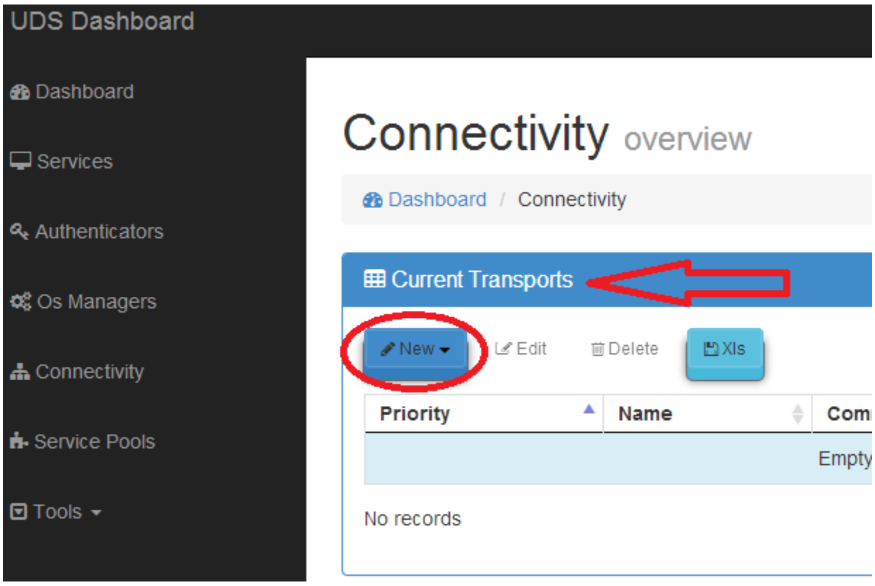
- HTML5 RDP Transport
- NX Transport (direct)
- NX Transport (tunneled)
- RDP Transport (direct)
- RDP Transport (tunneled)
- RGS Transport (direct)
- RGS Transport (tunneled)

We can configure the "Transport" indicated as "direct" for users accesses from an internal LAN, VPN, LAN Extension, etc...

We can configure the "Transport" indicated as "Tunneled" to user Access through a WAN. These "Transport" will be supported in the UDS Tunneler server to make the connection against the virtual desktops.

The HTML5 "Transport" can be used for any type of Access. This "Transport" uses the UDS Tunneler server to make the connection against the virtual desktops.

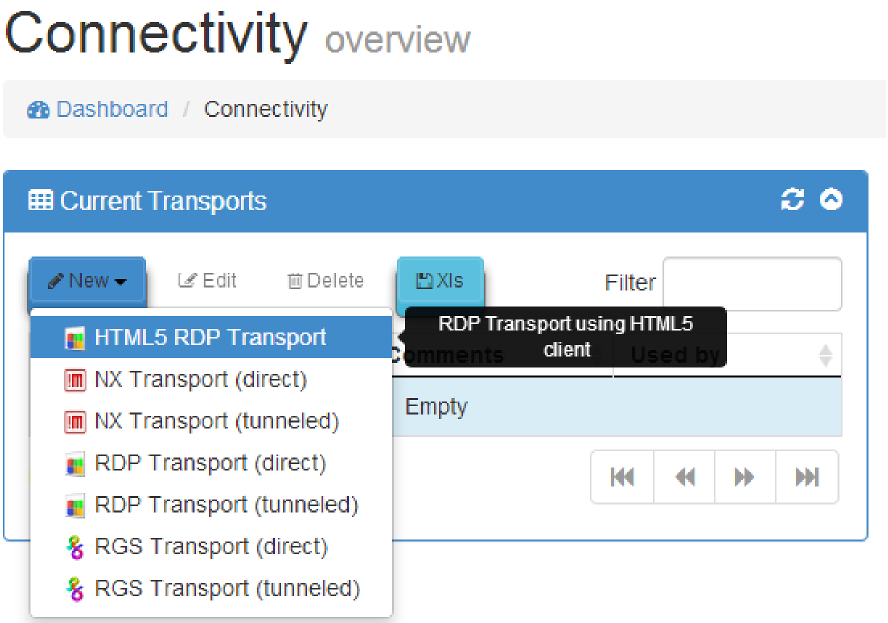
To create a "Transport", in the "Connectivity" section, click "New" in the section "Current Transports".



4.6.1 HTML5 RDP Transport

A "HTML5 RDP Transport" allows Access to Windows and Linux virtual desktops through RDP protocol using a browser which supports HTML5 (for Linux desktops the machines must have the XRDP packet installed. For Windows desktops the RDP access need to be set up).

This transport uses UDS Tunneler server to make the connection against the virtual desktops. It has to be configured beforehand in order to work properly.



Connectivity overview

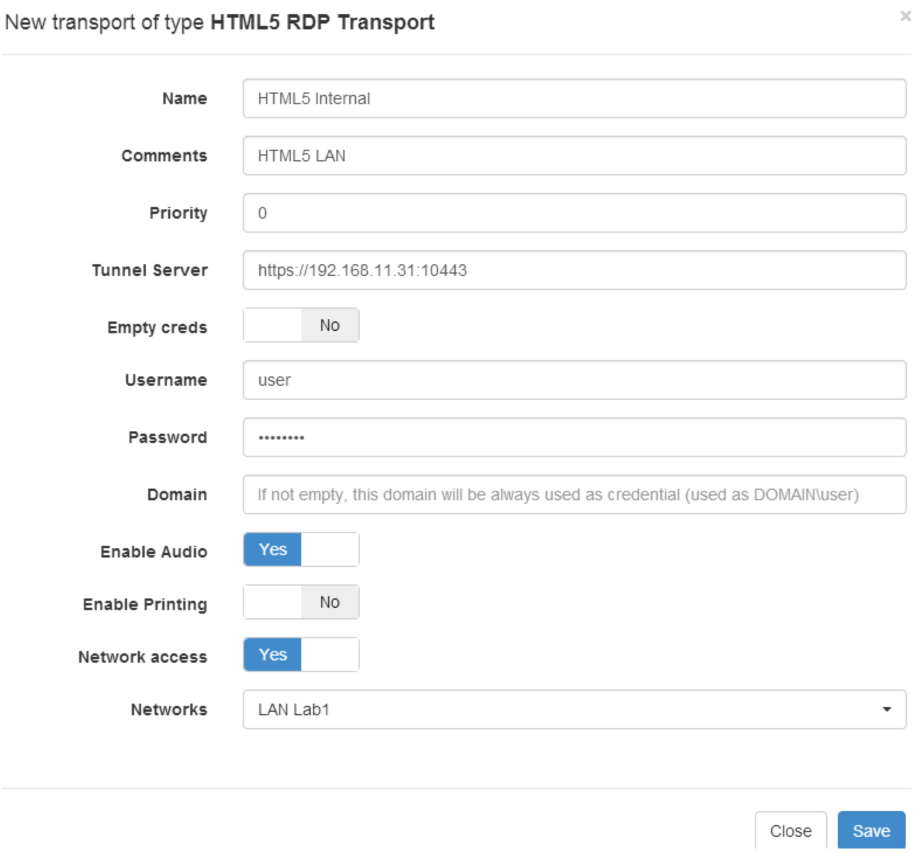
Dashboard / Connectivity

Current Transports

For HTML5 RDP Transport, we configure the name of the transport, the IP address of the UDS tunneler server and port (“Tunnel Server” field) with the format https://IP_Tunneler:10443 (port by default). We indicate whether we want to redirect specific credentials to the virtual desktop, and if the “Empty creds” box is checked, no credential will pass through to the virtual desktop. If we do not check the “Empty creds” box and enter a “username” and “password” (it is also possible to enter a domain name), these credentials will be redirected to the virtual desktops.

In the "Network access" section, we indicate if in the selected network in "Networks" access to users through this “Transport” will be allowed (the available networks will be the configured ones in the "Networks" section).

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values).



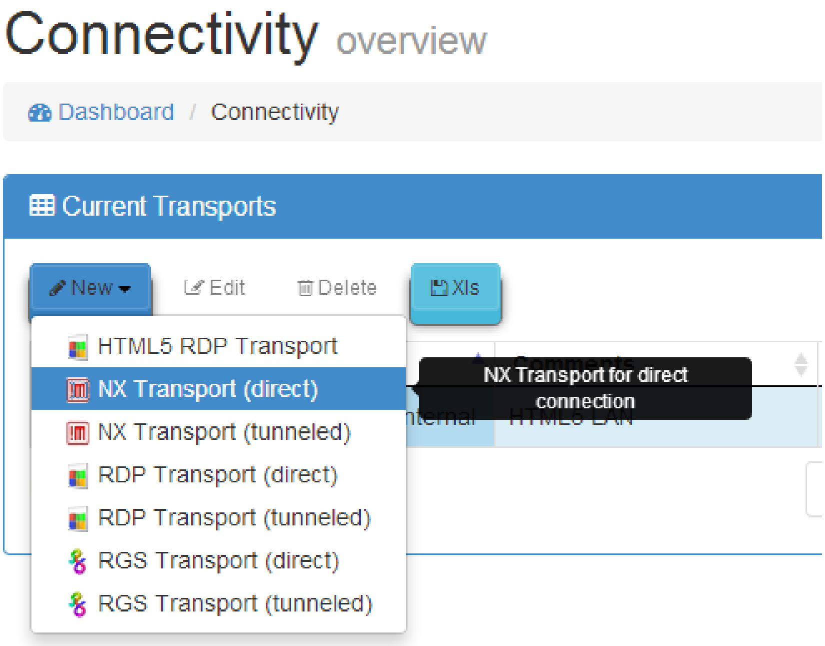
New transport of type **HTML5 RDP Transport**

Name	HTML5 Internal
Comments	HTML5 LAN
Priority	0
Tunnel Server	https://192.168.11.31:10443
Empty creds	No
Username	user
Password	••••••••
Domain	If not empty, this domain will be always used as credential (used as DOMAIN\user)
Enable Audio	Yes
Enable Printing	No
Network access	Yes
Networks	LAN Lab1

4.6.2 NX Transport (direct)

A "NX Transport (direct)" allows Access to Linux virtual desktops through NX software (the virtual machines and the connection clients must have NX installed).

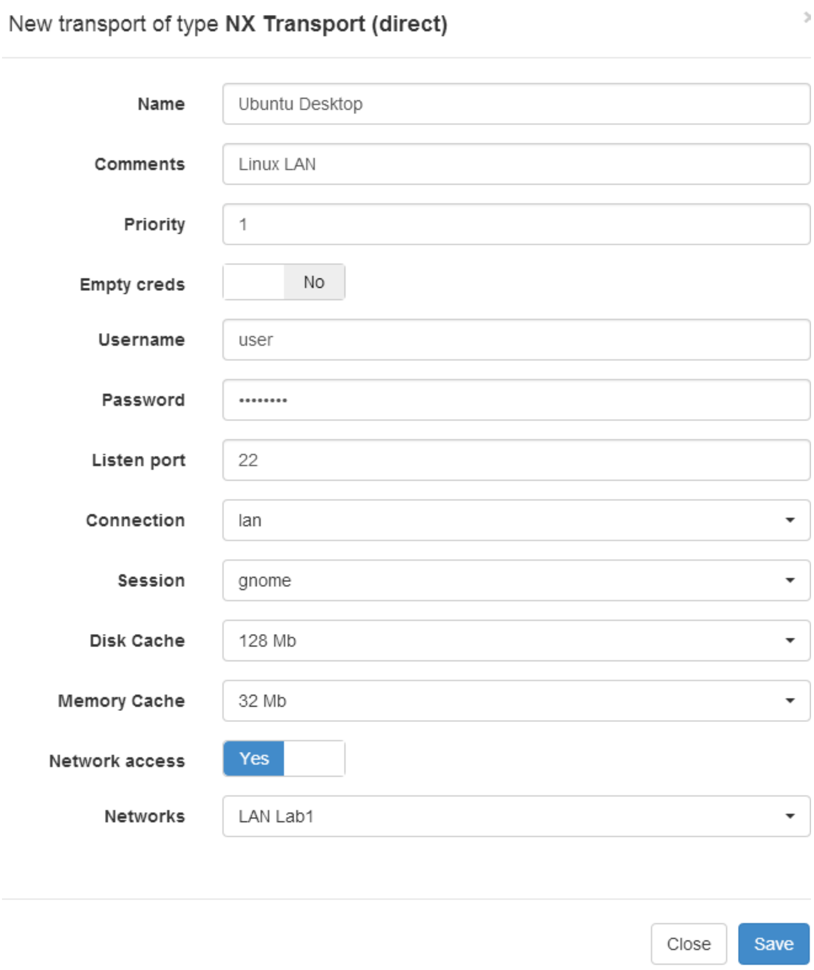
Currently, the NX supported version is 3.5



For a NX Transport (direct), we configure the transport name, we indicate whether we want to redirect specific credentials to the virtual desktop. If the “Empty creds” box is checked, no credential will pass through to the virtual desktop. If we do not check the “Empty creds” box and enter a “username” and “password”, these credentials will be redirected to the virtual desktops.

In the "Network access" section, we indicate if in the selected network in "Networks" access to users through this “Transport” will be allowed (the available networks will be the configured ones in the "Networks" section).

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values). We can also indicate what port and optimization connection parameters we want to use for the connection, such as: "Connection", "Session", "Disk Cache" and "Memory Cache".



New transport of type **NX Transport (direct)**

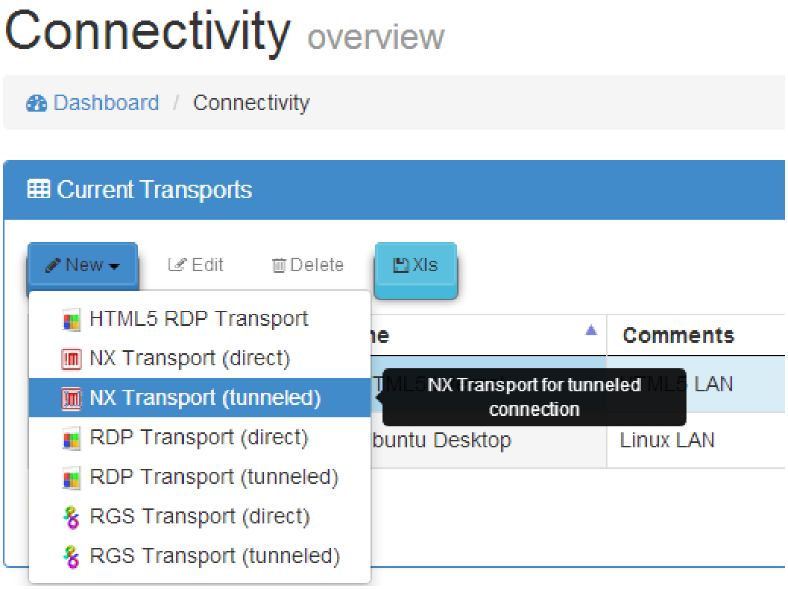
Name	Ubuntu Desktop
Comments	Linux LAN
Priority	1
Empty creds	No
Username	user
Password	••••••••
Listen port	22
Connection	lan
Session	gnome
Disk Cache	128 Mb
Memory Cache	32 Mb
Network access	Yes
Networks	LAN Lab1

4.6.3 NX Transport (tunneled)

A "NX Transport (tunneled)" allows Access to Linux virtual desktops through NX software (the virtual machines and the connection clients must have NX installed).

Currently, the NX supported version is 3.5

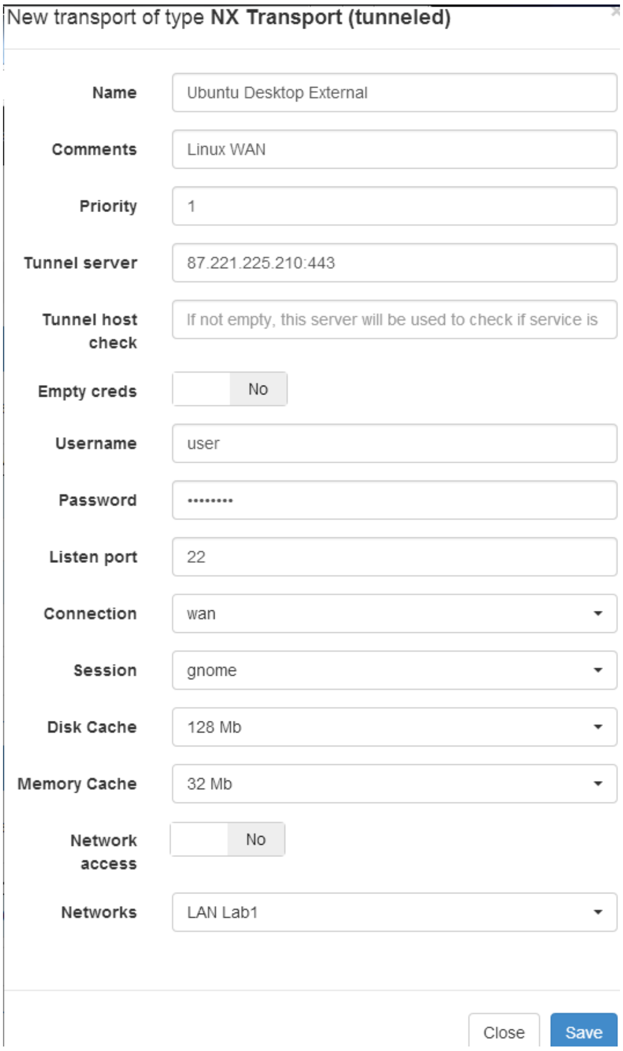
This transport uses UDS tunneler server to make the connection against the virtual desktops, and it needs to be configured beforehand in order to work properly.



For a NX Transport (tunneled), we configure the transport name, the UDS Tunneler server IP address and a port ("tunnel server" field) with the format IP_Tunneler:443 (port by default). We indicate if we want to redirect specific credentials to the virtual desktop. If we check the "Empty creds" box, no credential will be passed to the virtual desktop. If we don't check the "Empty creds" box and we indicate a "username" and "password", these credentials will be redirected to the virtual desktop.

In the "Network access" section, we indicate if in the selected network in "Networks" access to users through this "Transport" will be allowed (the available networks will be the configured ones in the "Networks" section).

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values). We can also indicate what port and optimization connection parameters we want to use for the connection, such as: "Connection", "Session", "Disk Cache" and "Memory Cache".

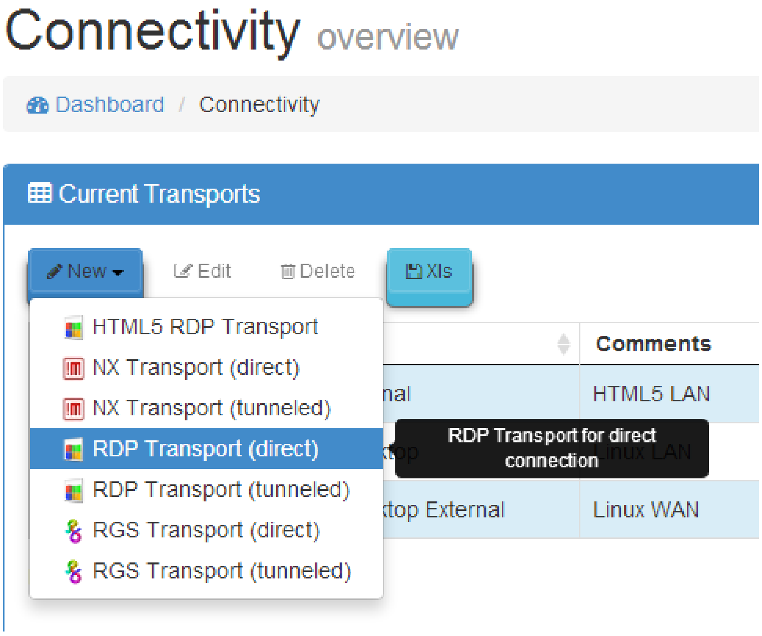


New transport of type **NX Transport (tunneled)**

Name	Ubuntu Desktop External
Comments	Linux WAN
Priority	1
Tunnel server	87.221.225.210:443
Tunnel host check	If not empty, this server will be used to check if service is
Empty creds	No
Username	user
Password	••••••••
Listen port	22
Connection	wan
Session	gnome
Disk Cache	128 Mb
Memory Cache	32 Mb
Network access	No
Networks	LAN Lab1

4.6.4 RDP Transport (direct)

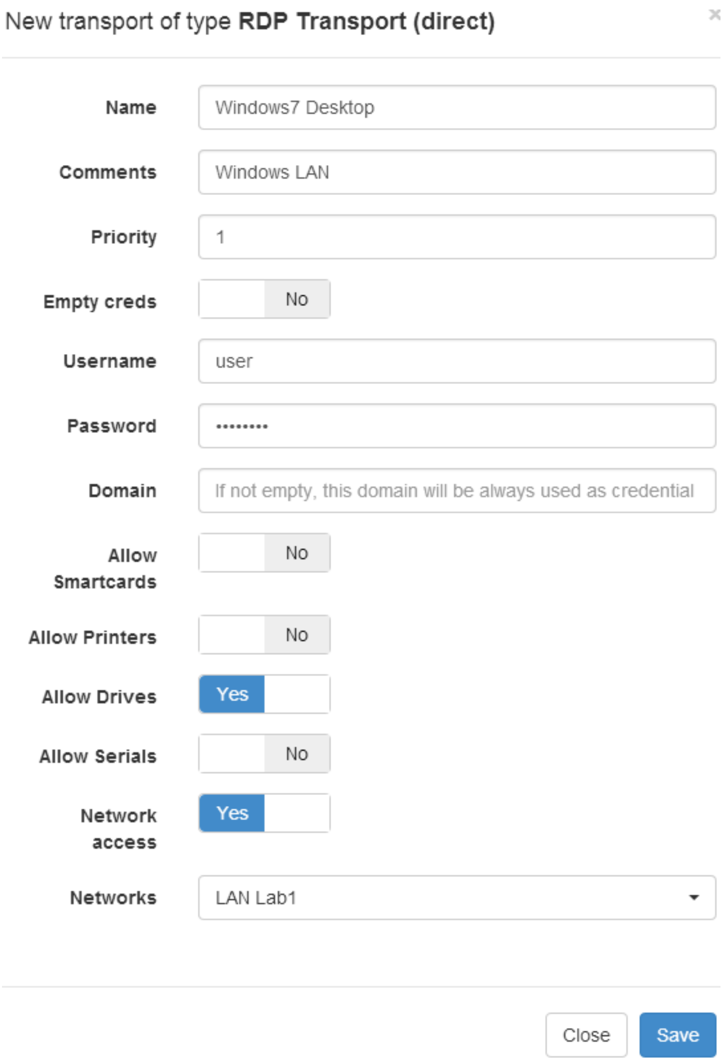
A "RDP Transport (direct)" allows access to Windows virtual desktops through RDP protocol (the virtual machines must have RDP service enabled).



For the RDP Transport (direct), we configure the transport name, we indicate whether we want to redirect specific credentials to the virtual desktop. If the "Empty creds" box is checked, no credential will pass through to the virtual desktop. If we do not check the "Empty creds" box and enter a "username" and "password" (it is also possible to enter a domain name), these credentials will be redirected to the virtual desktop.

In the "Network access" section, we indicate if in the selected network in "Networks" access to users through this "Transport" will be allowed (the available networks will be the configured ones in the "Networks" section).

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values). We can also indicate the parameters "Allow Smartcards", "Allow Printers", "Allow Drives" and "Allow Serials".

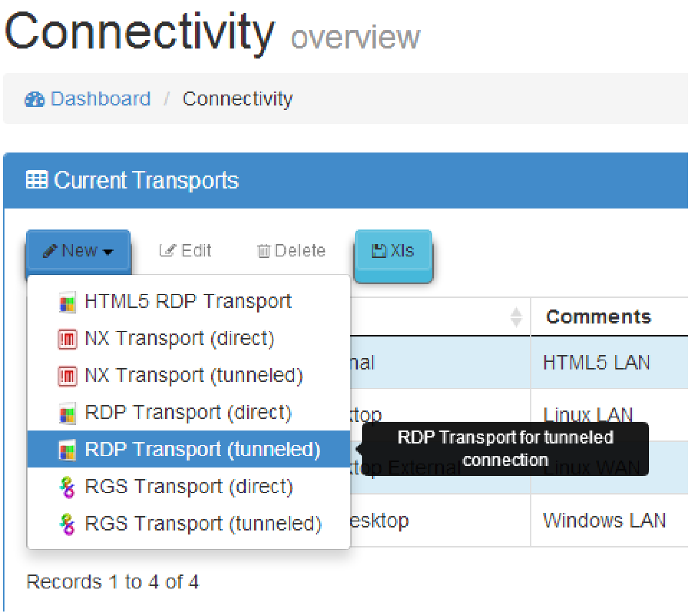

New transport of type **RDP Transport (direct)**

Name	Windows7 Desktop
Comments	Windows LAN
Priority	1
Empty creds	No
Username	user
Password	••••••••
Domain	If not empty, this domain will be always used as credential
Allow Smartcards	No
Allow Printers	No
Allow Drives	Yes
Allow Serials	No
Network access	Yes
Networks	LAN Lab1

4.6.5 RDP Transport (tunneled)

A "RDP Transport (tunneled)" allows access to Windows virtual desktops through RDP protocol (the virtual machines must have RDP service enabled).

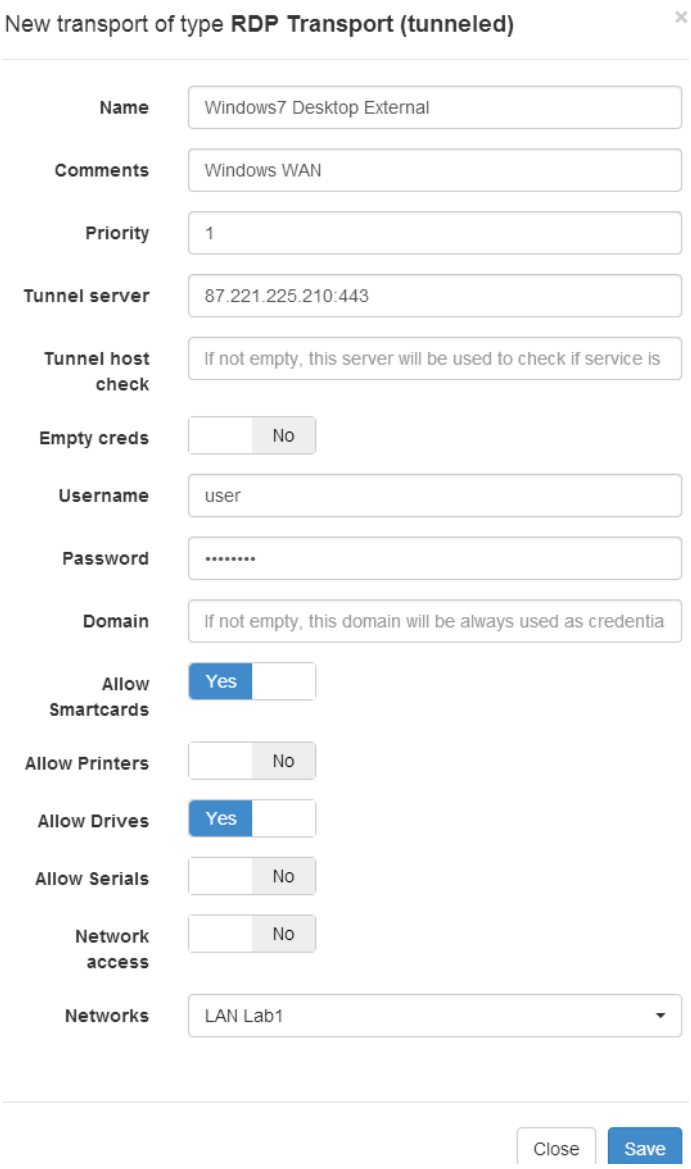
This transport uses UDS Tunneler server to make the connection against the virtual desktops. It must be configured beforehand in order to work properly.



For the RDP transport (tunneled), we configure the transport name, the IP address of the UDS tunneler server and port (“Tunnel server” field) with the format: IP_Tunneler:443 (port by default). We indicate whether we want to redirect specific credentials to the virtual desktop, and if the “Empty creds” box is checked, no credential will pass through to the virtual desktop. If we do not check the “Empty creds” box and enter a “username” and “password” (it is also possible to enter a user domain), these credentials will be redirected to the virtual desktop.

In the "Network access" section, we indicate if in the selected network in "Networks" access to users through this “Transport” will be allowed (the available networks will be the configured ones in the "Networks" section).

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values). We can also indicate the parameters "Allow Smartcards", "Allow Printers", "Allow Drives" and "Allow Serials".

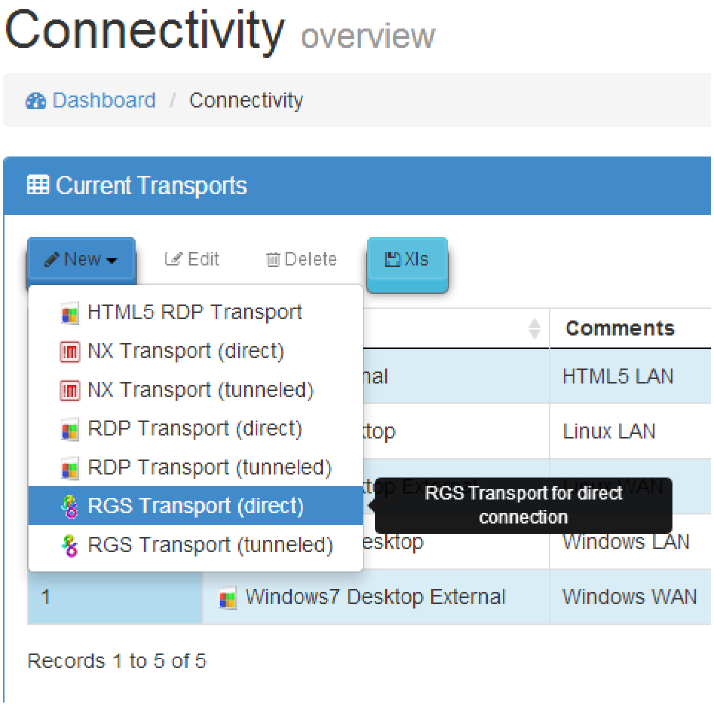


New transport of type **RDP Transport (tunneled)**

Name	Windows7 Desktop External
Comments	Windows WAN
Priority	1
Tunnel server	87.221.225.210:443
Tunnel host check	If not empty, this server will be used to check if service is
Empty creds	No
Username	user
Password	••••••••
Domain	If not empty, this domain will be always used as credentia
Allow Smartcards	Yes
Allow Printers	No
Allow Drives	Yes
Allow Serials	No
Network access	No
Networks	LAN Lab1

4.6.6 RGS Transport (direct)

A "RGS Transport (direct)" allows access to Windows virtual desktops through RGS protocol (the virtual machines and the connection clients must have RGS service installed).



Connectivity overview

Dashboard / Connectivity

Current Transports

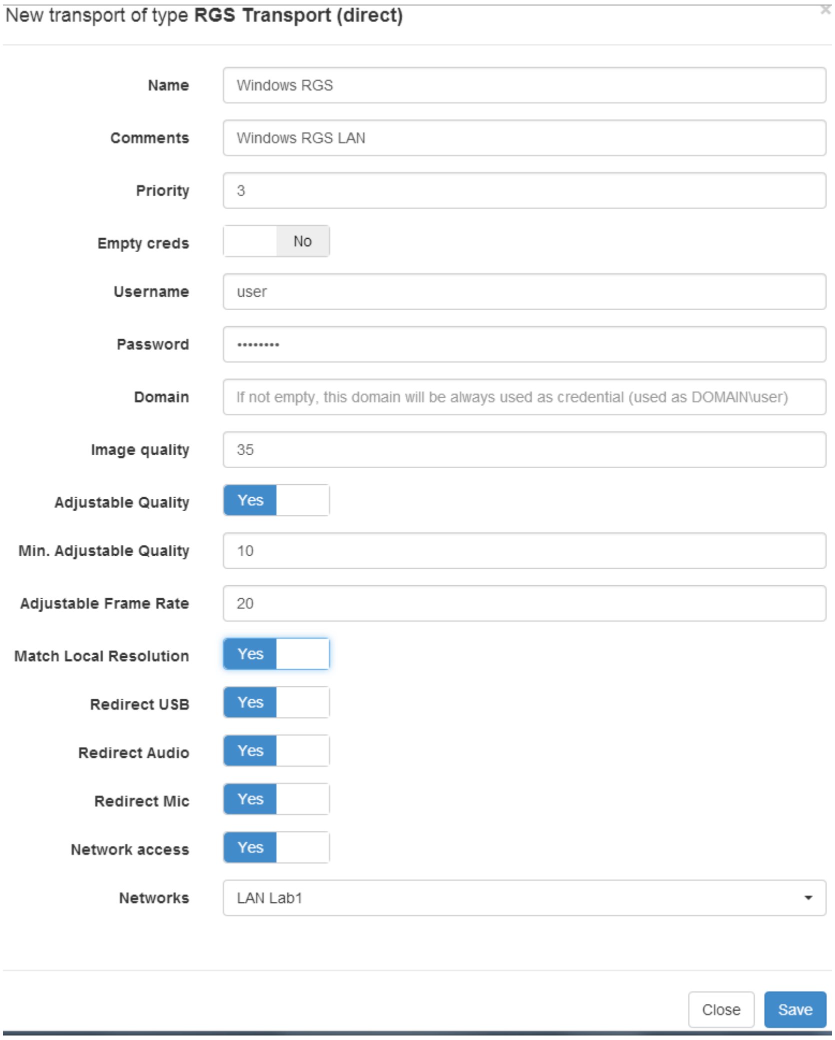
		Comments
	nal	HTML5 LAN
	ktop	Linux LAN
	ktop Ex	WAN
	esktop	Windows LAN
1	Windows7 Desktop External	Windows WAN

Records 1 to 5 of 5

For the RGS transport (direct), we configure the transport name and we indicate whether we want to redirect specific credentials to the virtual desktop. If the “Empty creds” box is checked, no credential will pass through to the virtual desktop. If we do not check the “Empty creds” box and enter a “username” and “password” (it is also possible to enter a user domain), these credentials will be redirected to the virtual desktop.

In the "Network access" section, we indicate if in the selected network in "Networks" access to users through this “Transport” will be allowed (the available networks will be the configured ones in the "Networks" section).

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values). We can also indicate the parameters “Image Quality”, “Adjustable Quality”, “Min Adjustable Quality”, “Adjustable Frame Rate”, “Match Local Resolution”, “Redirect USB”, “Redirect Audio” and Redirect Mic”.



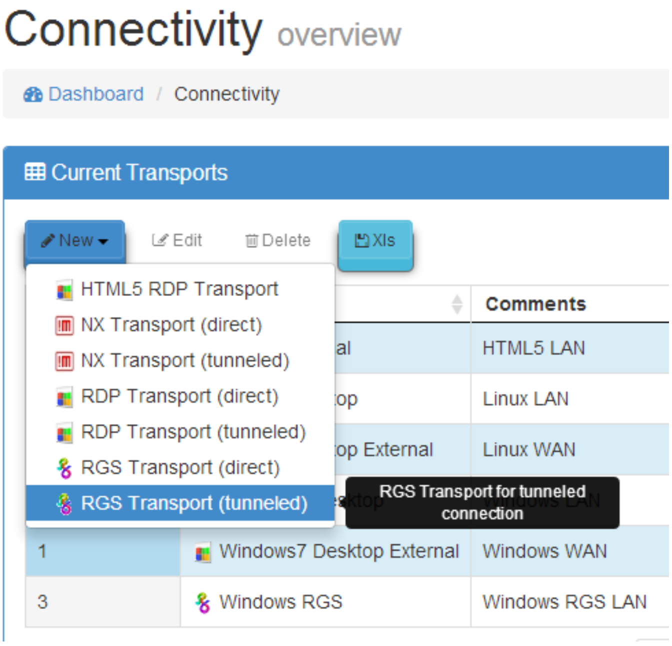
New transport of type **RGS Transport (direct)**

Name	Windows RGS
Comments	Windows RGS LAN
Priority	3
Empty creds	No
Username	user
Password	••••••••
Domain	If not empty, this domain will be always used as credential (used as DOMAIN\user)
Image quality	35
Adjustable Quality	Yes
Min. Adjustable Quality	10
Adjustable Frame Rate	20
Match Local Resolution	Yes
Redirect USB	Yes
Redirect Audio	Yes
Redirect Mic	Yes
Network access	Yes
Networks	LAN Lab1

4.6.7 RGS Transport (tunneled)

A "RGS Transport (tunneled)" allows access to Windows virtual desktops through RGS protocol (the virtual machines and the connection clients must have RGS service installed).

This transport uses UDS Tunneler server to make the connection against the virtual desktops. It must be configured beforehand in order to work properly.

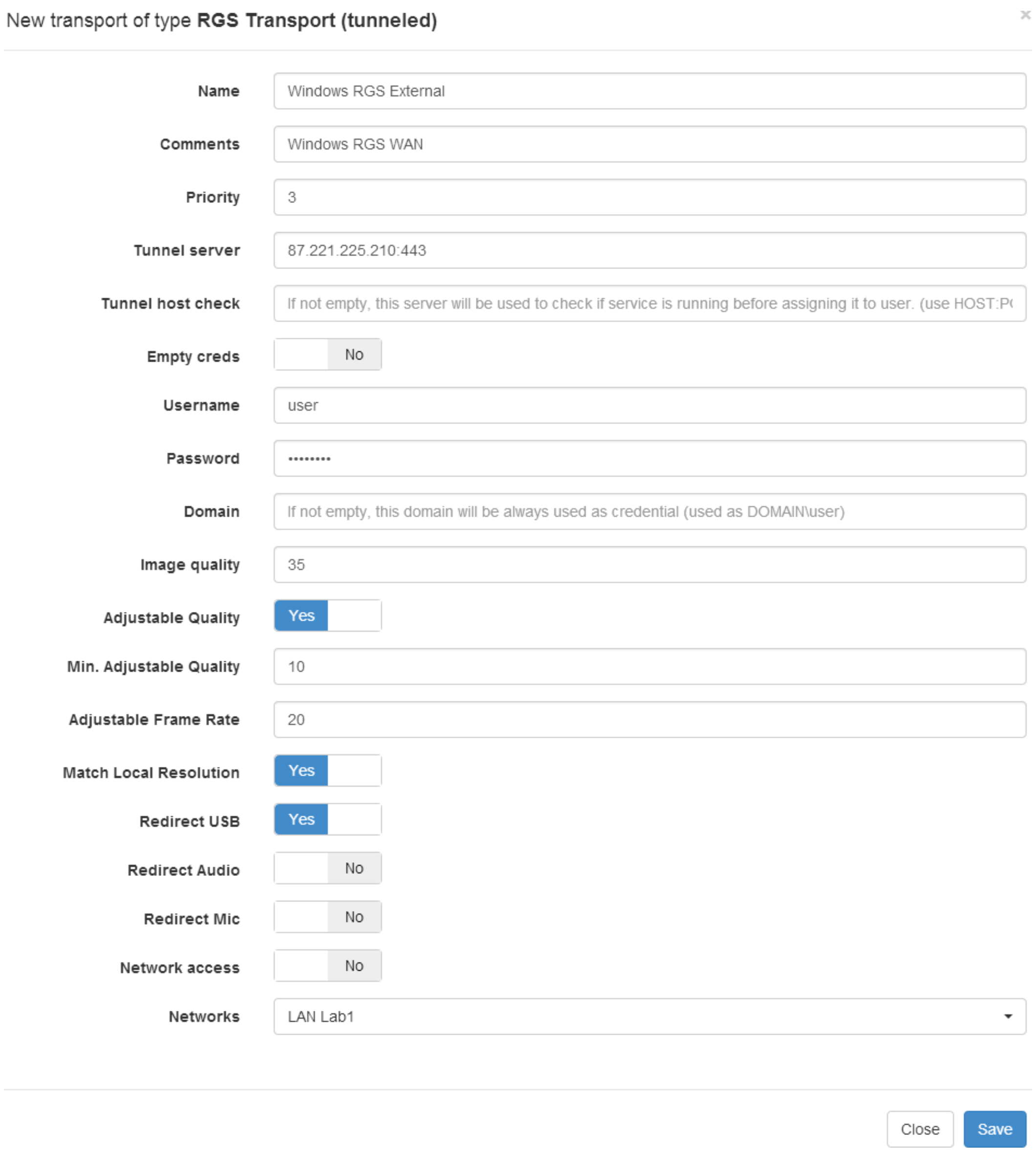


		Comments
	al	HTML5 LAN
	op	Linux LAN
	op External	Linux WAN
	esktop	ows LAN
1	Windows7 Desktop External	Windows WAN
3	Windows RGS	Windows RGS LAN

For the RGS Transport (tunneled), we configure the transport name, the IP address of the UDS Tunneler server and port (“Tunnel server” field) with the format IP_Tunneler:443 (port by default). We indicate whether we want to redirect specific credentials to the virtual desktop, and if the “Empty creds” box is checked, no credential will pass through to the virtual desktop. If we do not check the “Empty creds” box and enter a “username” and “password” (it is also possible to enter a user domain), these credentials will be redirected to the virtual desktop.

In the "Network access" section, we indicate if in the selected network in "Networks" access to users through this “Transport” will be allowed (the available networks will be the configured ones in the "Networks" section).

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values). We can also indicate the parameters "Image Quality", "Adjustable Quality", "Min Adjustable Quality", "Adjustable Frame Rate", "Match Local Resolution", "Redirect USB", "Redirect Audio" and Redirect Mic".



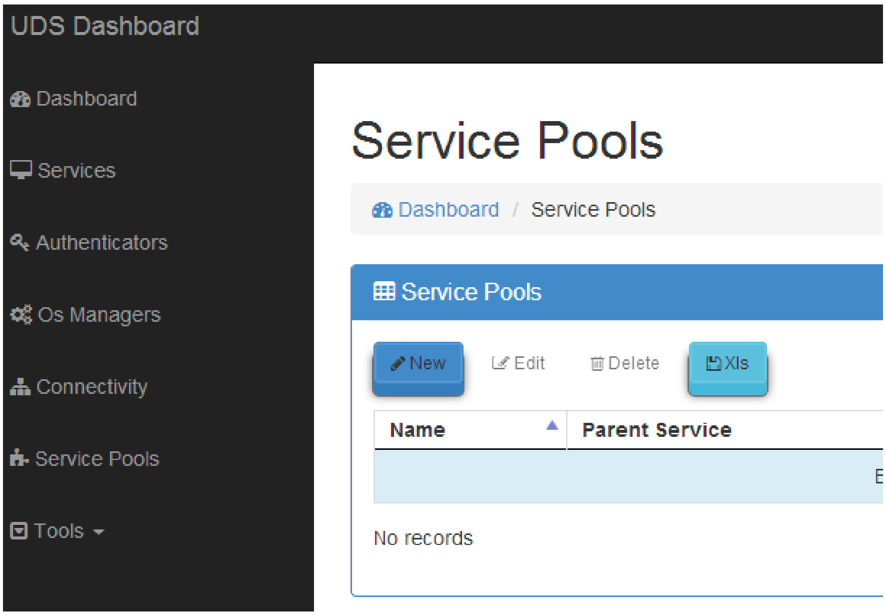
New transport of type **RGS Transport (tunneled)**

Name	Windows RGS External
Comments	Windows RGS WAN
Priority	3
Tunnel server	87.221.225.210:443
Tunnel host check	If not empty, this server will be used to check if service is running before assigning it to user. (use HOST:P(
Empty creds	No
Username	user
Password	••••••••
Domain	If not empty, this domain will be always used as credential (used as DOMAIN\user)
Image quality	35
Adjustable Quality	Yes
Min. Adjustable Quality	10
Adjustable Frame Rate	20
Match Local Resolution	Yes
Redirect USB	Yes
Redirect Audio	No
Redirect Mic	No
Network access	No
Networks	LAN Lab1

4.7 Configuring “Services Pools”

Once the different pieces of UDS platform are configured, it is time to create a “Service Pools”. This will be made up by a “Base Service” created from a “Service Provider” and an “OS Manager”. We will have to indicate one or several “Transports”, one or several access “Network” (if no access network is specified, all networks will be allowed) and a group or groups of users to access this service.

To create a "Service Pools" click on "New".



To configure a “Service Pools” it is necessary to indicate:

Name: Service name which will be shown to the user to access the virtual desktop.

Base Services: Base service configured beforehand in a "Service Provider", where it will be used to make the virtual desktops based in “Linked Clones” deployment.

OS Manager: We will indicate an “OS Manager" created beforehand which configuration will be applied to each virtual desktop generated in this "Services Pool".

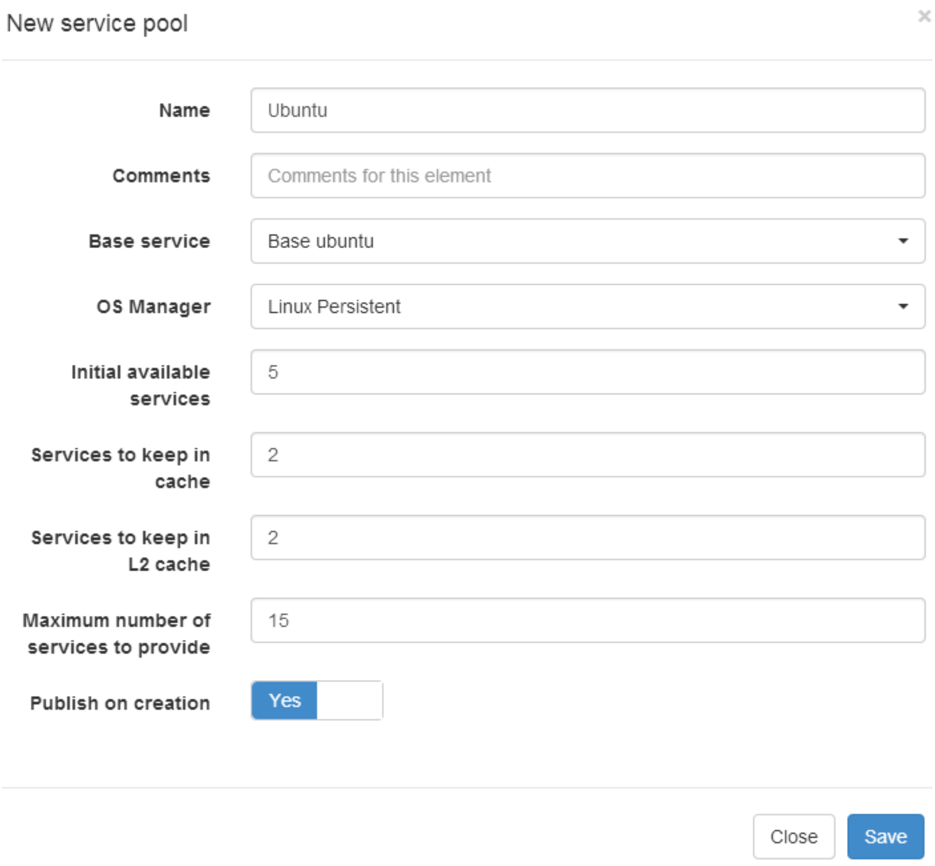
Initial available services: Virtual desktops that will be created, configured and will be initially in the system.

Services to keep in cache: Virtual desktops available in the system cache. These desktops will be configured and ready to be assigned (this number of desktops will be automatically generated until the maximum number of machines indicated in the field "Maximum number of services to provide" will be reached).

Services to keep in L2 cache: Virtual desktops in sleeping mode. These desktops will be configured and ready to be assigned when the system demands new desktops.

Maximum number of services to provide: Maximum number of virtual desktops created by UDS system in this "Service Pool".

Publish on creation: If this option is set up, the system will publish the new "Service Pool" when the user saves the creation of the new "Service Pool".

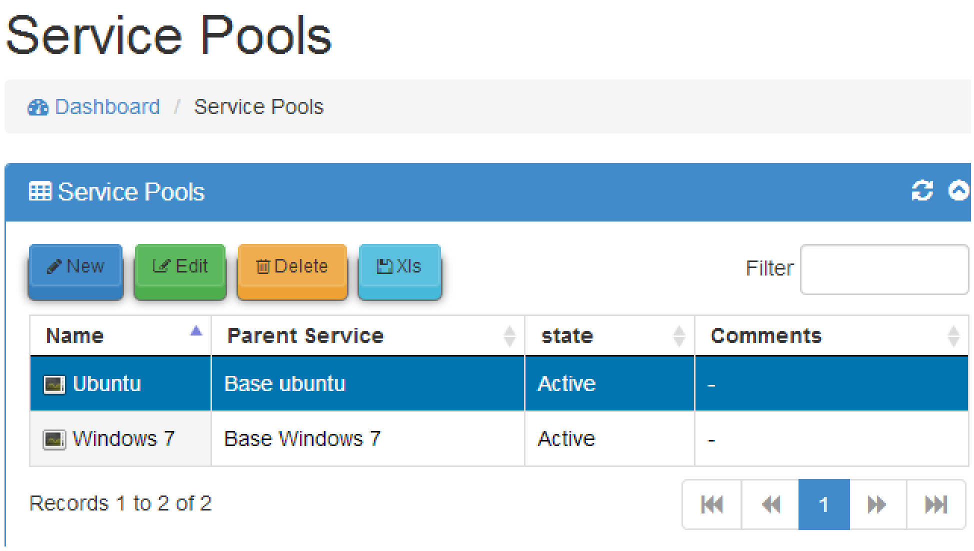


New service pool

Name	Ubuntu
Comments	Comments for this element
Base service	Base ubuntu
OS Manager	Linux Persistent
Initial available services	5
Services to keep in cache	2
Services to keep in L2 cache	2
Maximum number of services to provide	15
Publish on creation	Yes

When we save the creation of a "Service Pools" and if we have selected the option "Publish on creation", the system will start the publication of the service generating the base machine on which the virtual desktops will be deployed.

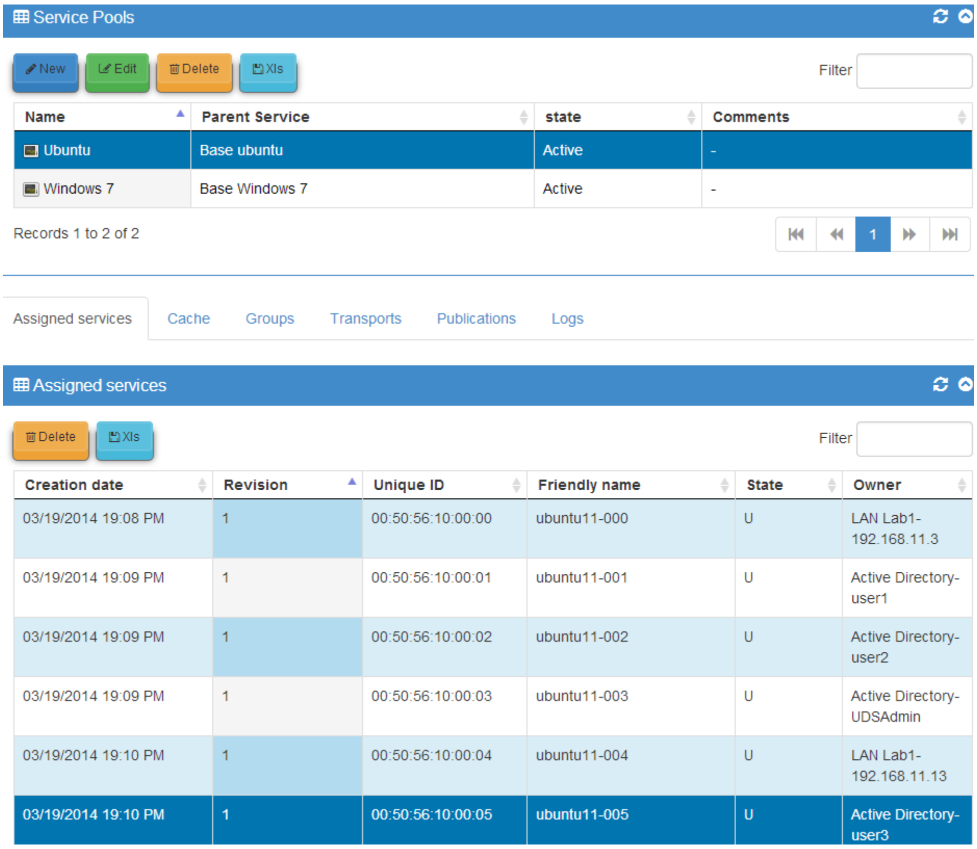
Clicking the "Delete" button, we will be able to delete a "Service Pool" and clicking “Edit" we will be able to change the name and all system cache values (Initial available services, Services to keep in cache, Services to keep in L2 cache and Maximum number of services to provide). But the "Base Service" and "OS Manager" can’t be modified.



Name	Parent Service	state	Comments
Ubuntu	Base ubuntu	Active	-
Windows 7	Base Windows 7	Active	-

Once a "Service Pool" is created, we select it and we'll have the following control and configuration menus available:

- **Assigned Services:** Virtual desktops assigned to users. It shows information about the desktop creation date, revision number (or publication) on which the desktop is generated, the MAC address of the VM network card, the virtual desktop DNS name, the current connection state, and the machine owner.



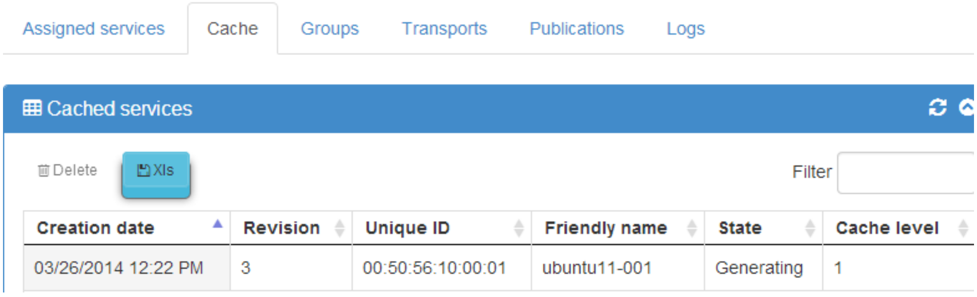
Name	Parent Service	state	Comments
Ubuntu	Base ubuntu	Active	-
Windows 7	Base Windows 7	Active	-

Records 1 to 2 of 2

Creation date	Revision	Unique ID	Friendly name	State	Owner
03/19/2014 19:08 PM	1	00:50:56:10:00:00	ubuntu11-000	U	LAN Lab1-192.168.11.3
03/19/2014 19:09 PM	1	00:50:56:10:00:01	ubuntu11-001	U	Active Directory-user1
03/19/2014 19:09 PM	1	00:50:56:10:00:02	ubuntu11-002	U	Active Directory-user2
03/19/2014 19:09 PM	1	00:50:56:10:00:03	ubuntu11-003	U	Active Directory-UDSAdmin
03/19/2014 19:10 PM	1	00:50:56:10:00:04	ubuntu11-004	U	LAN Lab1-192.168.11.13
03/19/2014 19:10 PM	1	00:50:56:10:00:05	ubuntu11-005	U	Active Directory-user3

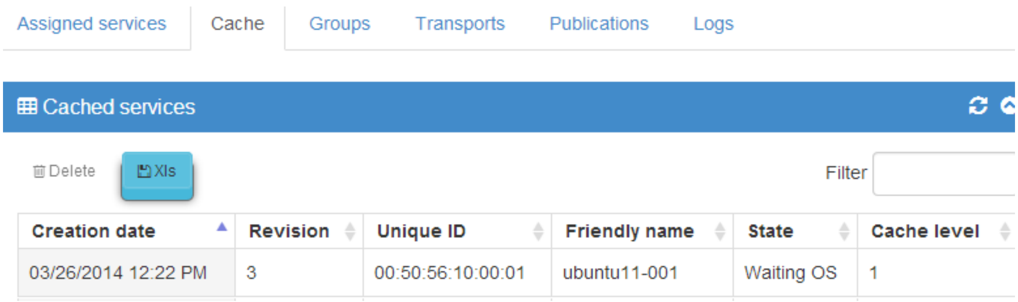
Selecting the virtual desktop and clicking on "Delete" we can delete it manually.

- **Cache**: Virtual desktops available in the system cache (including level 2 cache machines). These desktops will go through different states:
  - **Generating**: In this state the virtual desktops are being created in the virtualization platform.



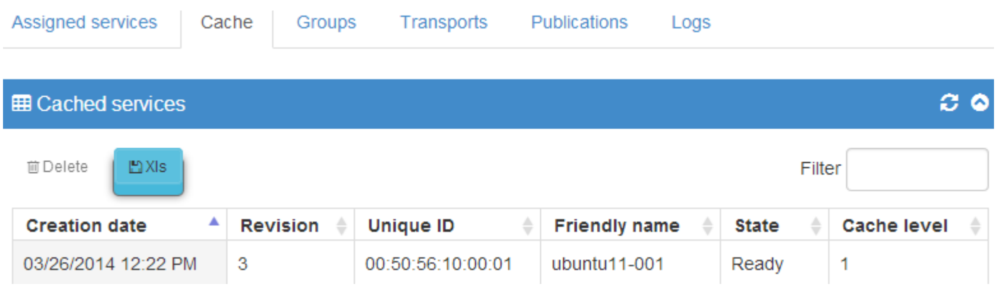
Creation date	Revision	Unique ID	Friendly name	State	Cache level
03/26/2014 12:22 PM	3	00:50:56:10:00:01	ubuntu11-001	Generating	1

  - **Waiting OS:** In this state the virtual desktops are being configurated with the parameters indicated in the “OS Manager”.



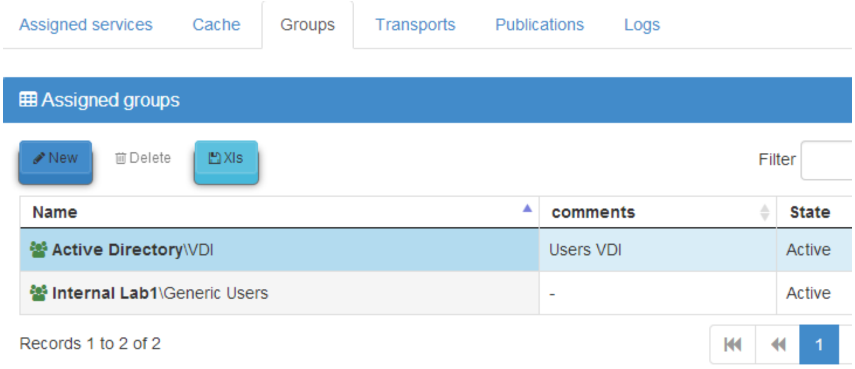
Creation date	Revision	Unique ID	Friendly name	State	Cache level
03/26/2014 12:22 PM	3	00:50:56:10:00:01	ubuntu11-001	Waiting OS	1

  - **Ready:** When a virtual desktop is in this state, it is available for use.



Creation date	Revision	Unique ID	Friendly name	State	Cache level
03/26/2014 12:22 PM	3	00:50:56:10:00:01	ubuntu11-001	Ready	1

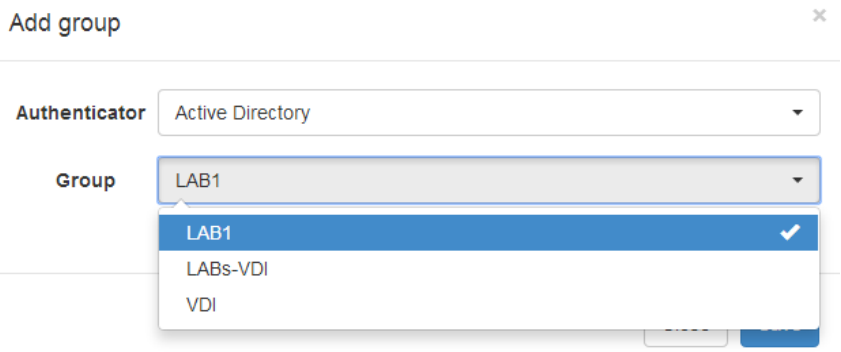
- **Groups:** To allow the users connection, it is necessary to assign access groups or metagroups. These groups or metagroups must be created in the "Authenticators" section and we will be able to assign one or several access groups or metagroups to each "Service Pools".



Name	comments	State
Active Directory\VDI	Users VDI	Active
Internal Lab1\Generic Users	-	Active

Records 1 to 2 of 2

We select the "Authenticator" and based on this choice we select the "Group Name".

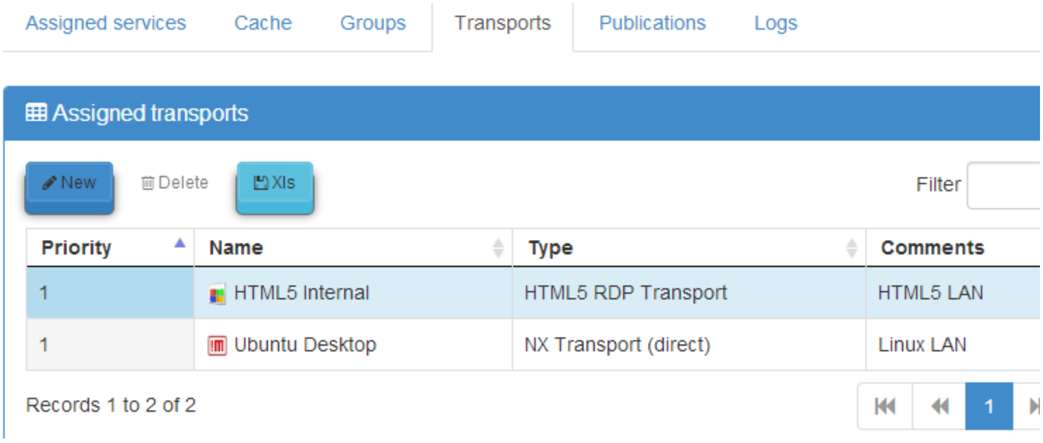


Add group

Authenticator: Active Directory

Group: LAB1

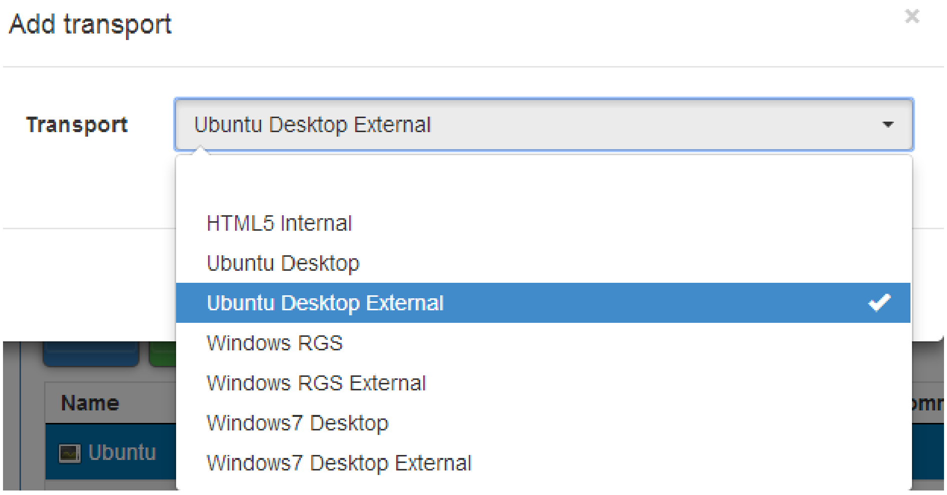
- **Transports:** The "Transport" to make the connection with the virtual desktop (beforehand added in the "Transports" section) will be indicated. The "Transport" with less priority will be configured by the system by default. For the other ones, the user will have to open the pull-down menu in the virtual desktops access window and select the one that corresponds.



Priority	Name	Type	Comments
1	HTML5 Internal	HTML5 RDP Transport	HTML5 LAN
1	Ubuntu Desktop	NX Transport (direct)	Linux LAN

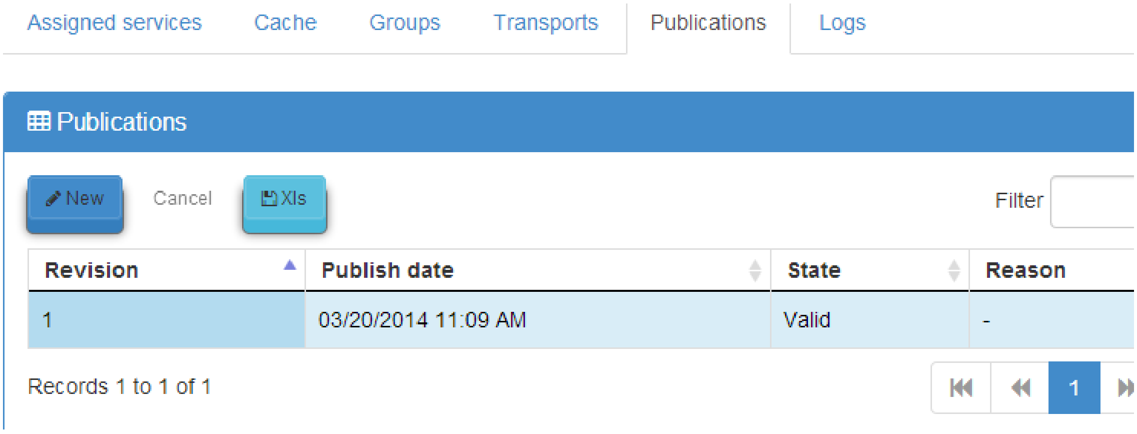
Records 1 to 2 of 2

Select the "Transport" we want to use in this "Service Pool" and save.



Transport: Ubuntu Desktop External

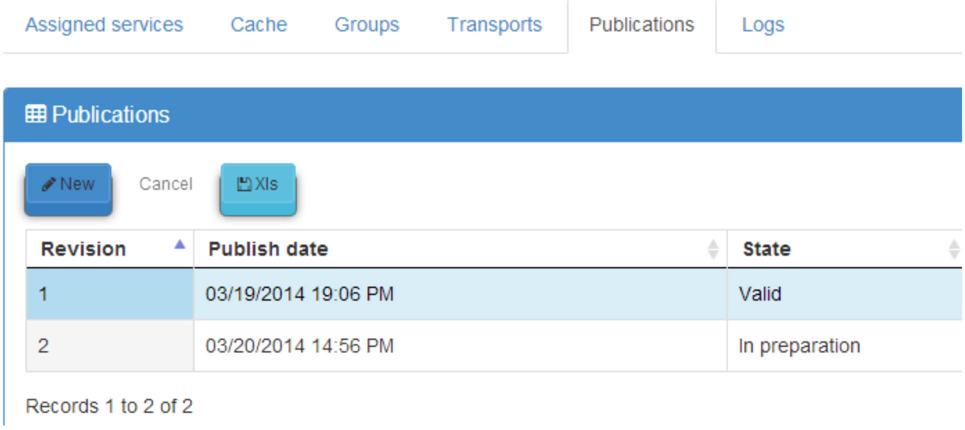
- **Publications:** From this menu we will be able to make a new service publication. Once the publication process has finished, the whole system cache with the new Linked Clones based on this last publication will be regenerated.



Revision	Publish date	State	Reason
1	03/20/2014 11:09 AM	Valid	-

Records 1 to 1 of 1

If we make a new publication, a new base machine will be generated and once it will be available, the system will delete the virtual desktops from the previous version and it will generate new ones based on the new publication.



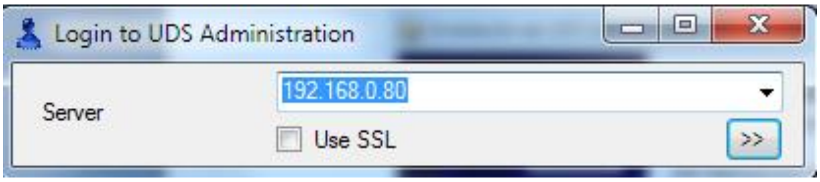
Revision	Publish date	State
1	03/19/2014 19:06 PM	Valid
2	03/20/2014 14:56 PM	In preparation

Records 1 to 2 of 2

5 ADMINISTERING UDS – UDS ADMINISTRATION CLIENT

Once the UDS platform has been installed, the system will be ready for its initial administering and configuration.

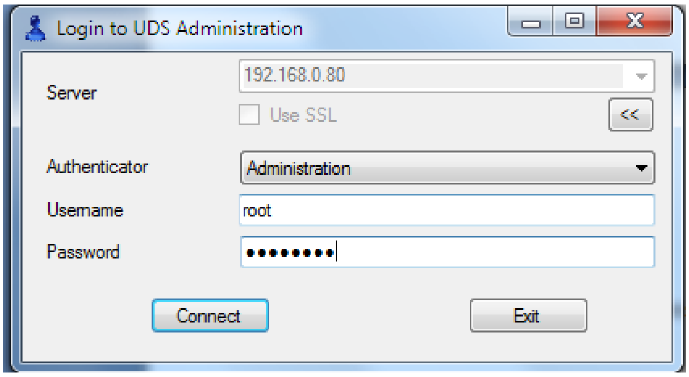
Open the administration client and indicate the network address of the UDS broker that is going to be managed.



Server: 192.168.0.80

☐ Use SSL

If it is the first time that you access the administration, select the "Administration" authenticator and enter the credentials provided in the UDS broker machine installation (point 3.2.1).



Server: 192.168.0.80

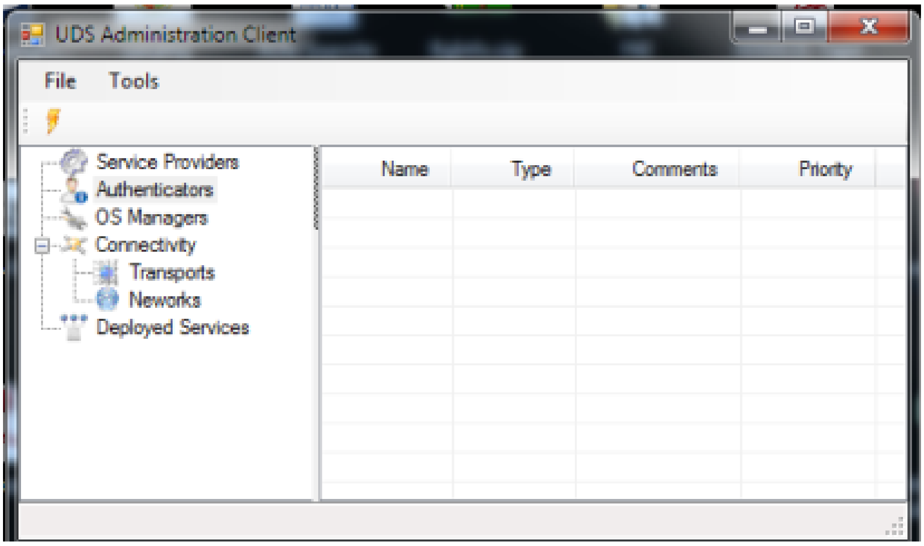
☐ Use SSL

Authenticator: Administration

Username: root

Password: ••••••••

Once inside the administration, the initial configuration of the services will begin.



The configuration of each one of the deployed services must be approached like the building of a puzzle, comprised of different elements:

- Each deployed service is made up of different elements or pieces (Base Services, OS Managers, Transports and Authenticators).
- Once the elements of the first deployed service have been configured, the creation thereof will begin, repeating the process with the next deployed service, if there is one.
- All of the configured deployed services together will form the type of virtual desktop deployment managed by the UDS platform.

5.1 Configuring "Service Providers"

A Service Provider is the organization responsible for offering IP services.

The services offered by UDS will be on-demand virtual desktops provided by a virtualization platform or persistent physical/virtual desktops assigned to specific users via assignment of IPs.

In order to build a deployed service, it is necessary to have created at least a service provider.

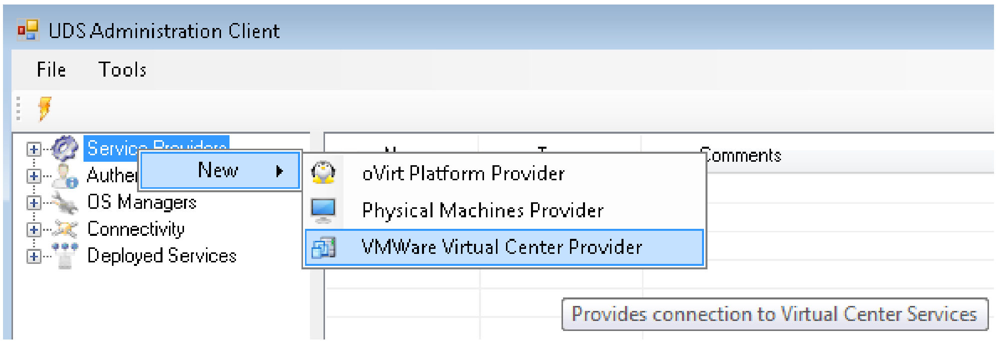
Currently, UDS allows the following service providers:

5.1.1 VDI platform with VMWare vSphere

Deployment of VDI platform via the VMware vSphere virtual infrastructure.

5.1.1.1 Registering "VMWare Virtual Center Provider" service provider

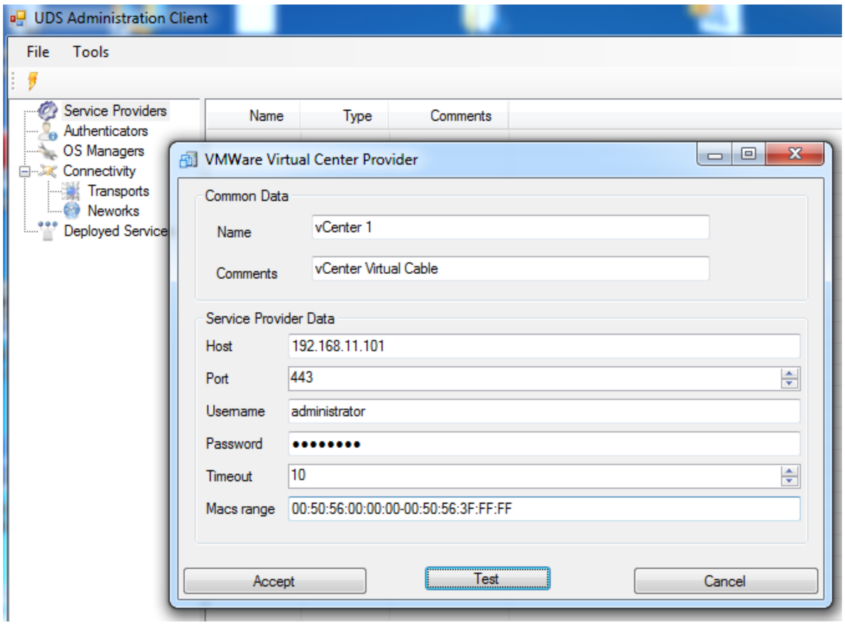
Select "VMware Virtual Center Provider".


UDS Administration Client
File
Tools
Service Providers
Authen
OS Managers
Connectivity
Deployed Services
New
oVirt Platform Provider
Physical Machines Provider
VMWare Virtual Center Provider
Comments
Provides connection to Virtual Center Services

In a VMware Virtual Center Provider, the minimum parameters to configure are: Service Name, vCenter server IP ("Host" field), a username and password with administrator rights on vCenter.

We can also select "Timeout" in the connection with vCenter and specify a range of MAC addresses for creating the virtual desktops.

By clicking the "Test" button, we can check if the connection has been made correctly.



Name: vCenter 1

Comments: vCenter Virtual Cable

Host: 192.168.11.101

Port: 443

Username: administrator

Password: ••••••••

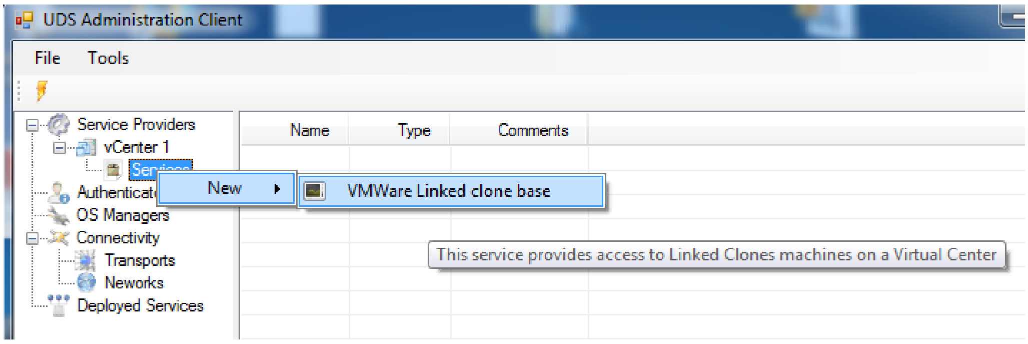
Timeout: 10

Macs range: 00:50:56:00:00:00-00:50:56:3F:FF:FF

Click Accept and we will already have a valid "Service Providers" to start creating base services in the VMware vCenter platform. We will be able to register all "VMware Virtual Center Provider" "Service Providers" we need in the UDS platform.

5.1.1.2 Configuring service based on “VMware Linked Clones”

Once the vSphere platform where the desktops will be created has been configured, you must create services base on VMWare Linked Clones.



This service provides access to Linked Clones machines on a Virtual Center

Choose a descriptive name for the template and configure the service parameters:

Datacenter: Datacenter where the service will be hosted.

Network: Network to which the desktops will be connected.

Pub. Resource Pool: vCenter resources Pool where the virtual desktops Linked Clones will be hosted (If there are no Pools in the VMware infrastructure, they will be created in the root).

Clones Folder: Location of the virtual desktops Linked Clones in the VM view and the vCenter templates.

Resource Pool: vCenter resources pool where the template to be used by the service is located.

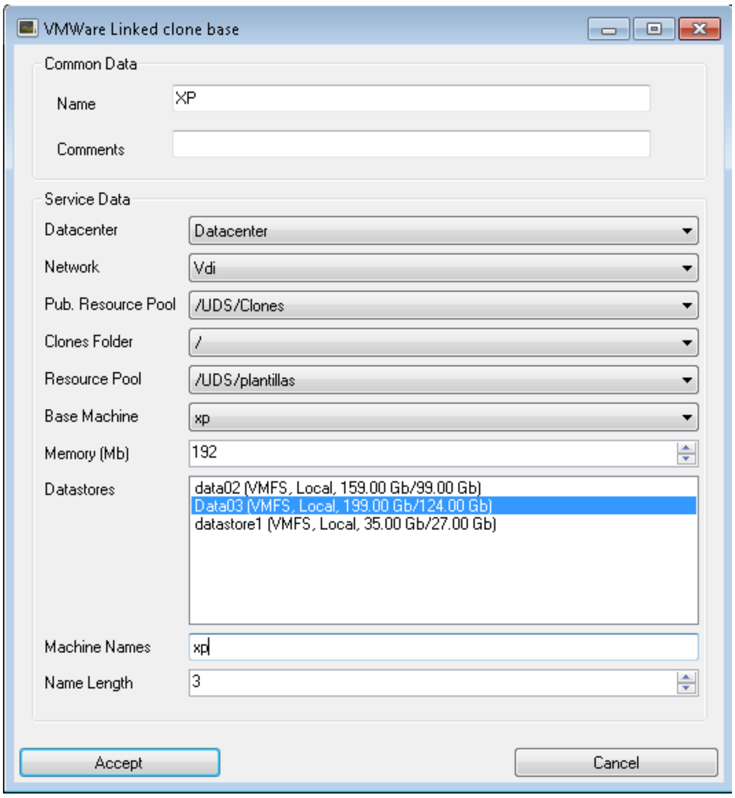
Base Machine: Template for deploying the virtual desktops.

Memory: Amount of memory to be assigned to the virtual desktops Linked Clones.

Datastores: Location where the publication of the service and the Linked Clones created will be stored. We can select one, several or all of the datastores by holding down the “Ctrl” button. If we select more than one, the system will always locate the new publications and desktops in the datastore with more free space available (by default, the system won’t generate new publications and it won’t create new virtual desktops in datastores with less than 30 GB of free space. This parameter can by modified in advanced options of UDS system).

Machine Names: Root name of all of the Linked Clones to be deployed on this service. (ex: Machine Names= XP).

Name Length: Number of digits of the counter attached to the root name of the desktops (ex: Name Length= 3, XP**001**...XP**999**).



VMWare Linked clone base

Common Data

Name: XP

Comments:

Service Data

Datacenter: Datacenter

Network: Vdi

Pub. Resource Pool: /UDS/Clones

Clones Folder: /

Resource Pool: /UDS/plantillas

Base Machine: xp

Memory (Mb): 192

Datastores:
- data02 (VMFS, Local, 159.00 Gb/99.00 Gb)
- Data03 (VMFS, Local, 199.00 Gb/124.00 Gb)
- datastore1 (VMFS, Local, 35.00 Gb/27.00 Gb)

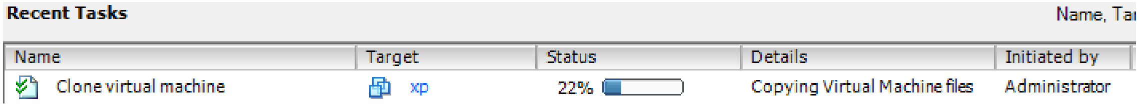
Machine Names: xp

Name Length: 3

Click Accept and we will already have a valid "VMWare Linked Clone Base" in the VMware vCenter platform. We will be able to register all "VMware Linked Clone Base" we need in the UDS platform.

Once the entire UDS environment has been configured and the first "Service Pools" has been created, we will be able to observe how the virtual desktops based on VMWare Linked Clones are deployed on the vCenter server.

The first task that the vCenter will perform will be to create a base machine (this machine will be generated every time we make a service publication), which will be a clone of the template selected when registering the service, with a hard drive size and characteristics equal to those of said template.

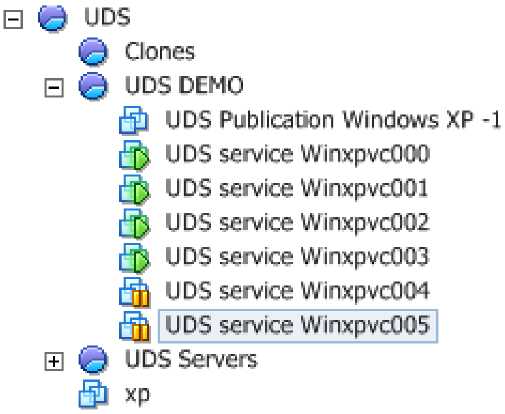


Recent Tasks Name, Tai

Name	Target	Status	Details	Initiated by
Clone virtual machine	xp	22%	Copying Virtual Machine files	Administrator

Once the process of creating the base machine has been completed (the UDS system calls it: "UDS Publication name_service –number_publication"), the creation of virtual desktops in the vCenter automatically begins (the UDS system calls it: "UDS service Machine_Name+Name_Length".

The hard drive space occupied by the virtual desktops ("linked clones") will be exclusively that which is occupied by the changes made by the users on the machines after their initial connection.



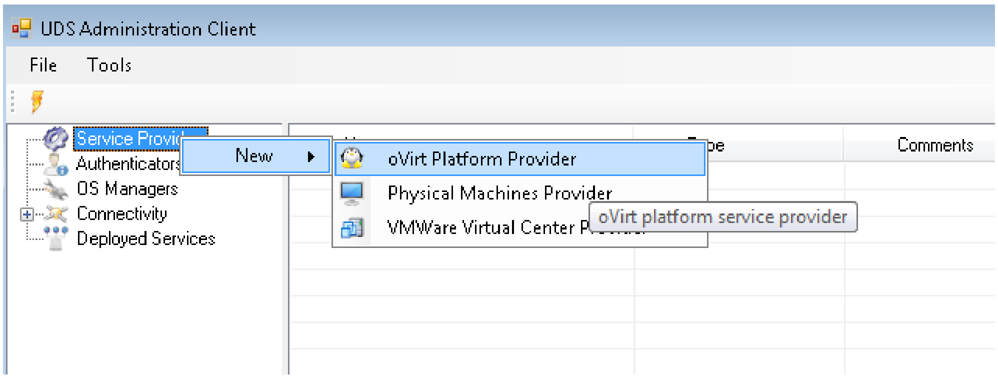
```
⊟ UDS
    Clones
  ⊟ UDS DEMO
      UDS Publication Windows XP -1
      UDS service Winxpvc000
      UDS service Winxpvc001
      UDS service Winxpvc002
      UDS service Winxpvc003
      UDS service Winxpvc004
      UDS service Winxpvc005
  ⊞ UDS Servers
    xp
```

5.1.2 VDI platform with oVirt

Deploying the VDI platform via the virtual oVirt infrastructure

5.1.2.1 Registration of service provider "ovirt Platform Provider"

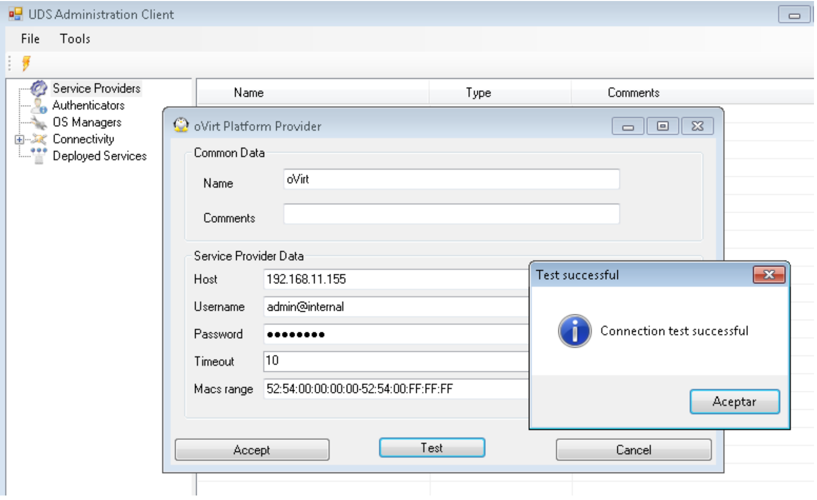
We select "oVirt Plataform Provider".



In an "oVirt Platform Provider," you must configure at least the following parameters: Service Name, oVirt-engine IP server ("Host" field), username (with user@domain format) and password with administration rights on the oVirt-engine.

We can also indicate the "Timeout" in the connection with oVirt-engine and specify a range of MAC addresses for creating the virtual desktops.

We will check that the connection has been correctly made by clicking on the "Test" button.



Common Data

Name: oVirt

Comments:

Service Provider Data

Host: 192.168.11.155

Username: admin@internal

Password: ●●●●●●●●

Timeout: 10

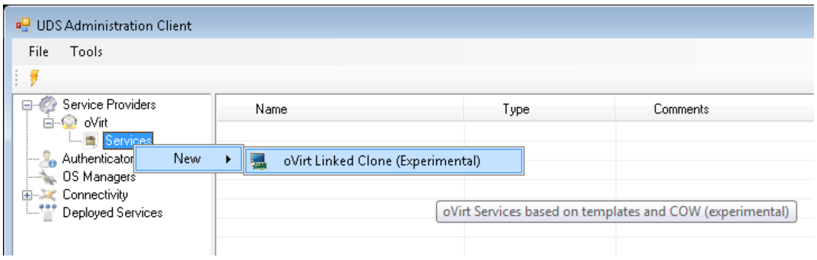
Macs range: 52:54:00:00:00:00-52:54:00:FF:FF:FF

Connection test successful

Click Accept and we will already have a valid "Service Providers" to start creating base services in the oVirt platform. We will be able to register all "oVirt Platform Provider" Service Providers we need in the UDS platform.

5.1.2.2 Configuring service based on "oVirt Linked Clone"

Once the oVirt platform where the desktops will be created has been configured, you must create "Base Service" based on oVirt Linked Clones.


UDS Administration Client
File
Tools
Service Providers
oVirt
Services
Authenticators
OS Managers
Connectivity
Deployed Services
New
oVirt Linked Clone (Experimental)
Name
Type
Comments
oVirt Services based on templates and COW (experimental)

Type a descriptive name for the template and the configure service parameters:

Base Machine: Virtual machine image from which the Linked Clones will be deployed.

Cluster: oVirt node cluster that will host the deployed Linked Clones.

Datastore Domain: Storage established for deploying the Linked Clones.

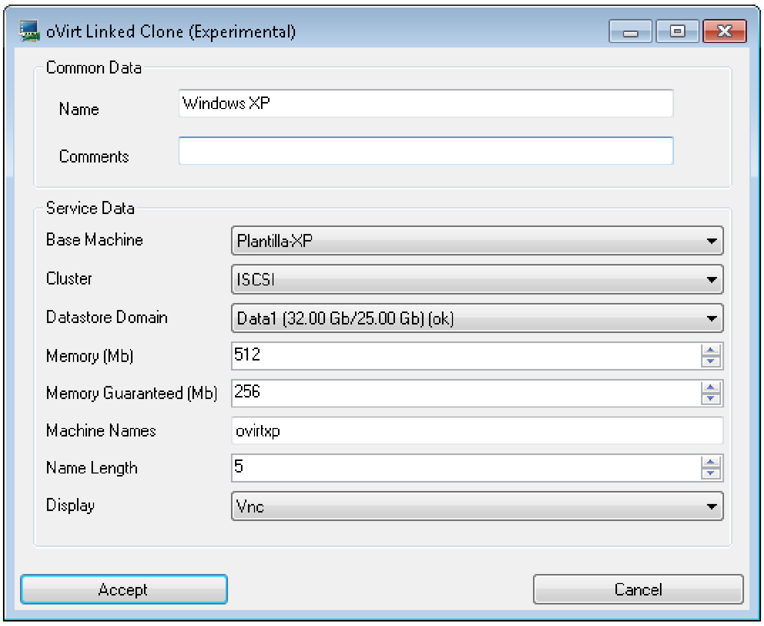
Memory: Amount of memory that will be assigned to the Linked Clones.

Memory Guaranteed: Amount of memory that will guarantee to the Linked Clones.

Machine Names: Root name of all of the Linked Clones to be deployed on this service (ex. Machine Names= ovirtxp).

Name Length: Number of counter digits attached to the root name of the desktops. (ex: Name Length= 5, ovirtxp**00001**...ovirtxp**99999**).

Display: Connection protocol to the virtual desktops deployed via Linked Clones.


oVirt Linked Clone (Experimental)

Common Data

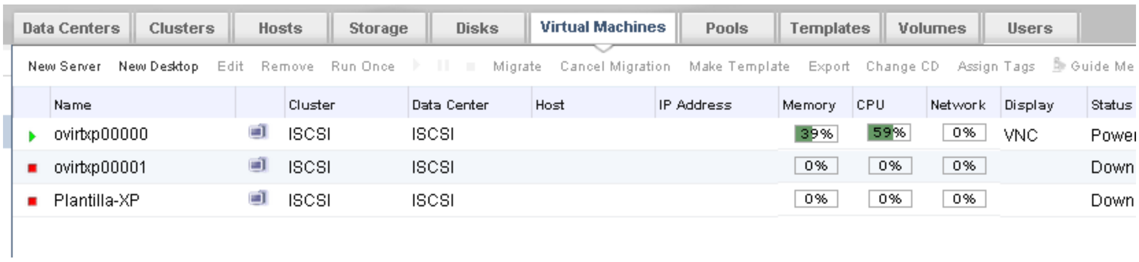
Name	Windows XP
Comments	

Service Data

Base Machine	Plantilla-XP
Cluster	ISCSI
Datastore Domain	Data1 (32.00 Gb/25.00 Gb) (ok)
Memory (Mb)	512
Memory Guaranteed (Mb)	256
Machine Names	ovirtxp
Name Length	5
Display	Vnc

Click Accept and we will already have a valid "oVirt Linked Clone" in the oVirt platform. We will be able to register all "oVirt Linked Clone" we need in the UDS platform.

When the entire UDS environment has been configured and the first "Service Pools" has been created, we will be able to observe how the virtual desktops based on oVirt Linked Clones are deployed on the oVirt-engine server.



	Name		Cluster	Data Center	Host	IP Address	Memory	CPU	Network	Display	Status
▶	ovirtxp00000		ISCSI	ISCSI			39%	59%	0%	VNC	Powe
■	ovirtxp00001		ISCSI	ISCSI			0%	0%	0%		Down
■	Plantilla-XP		ISCSI	ISCSI			0%	0%	0%		Down

5.1.3 Connection to persistent hardware

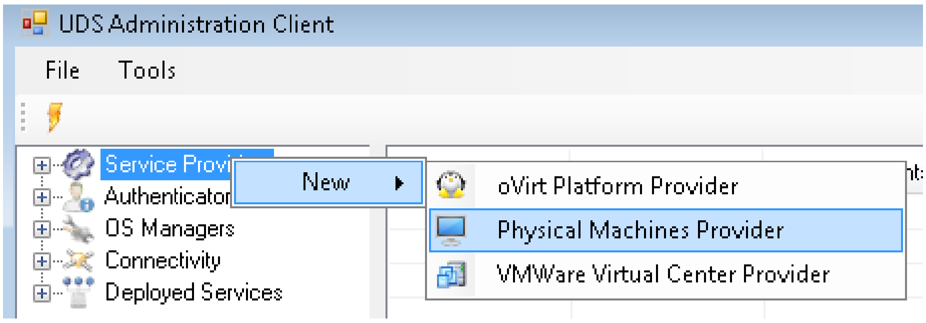
Access persistent hardware by assigning fixed-user IP addresses.

Assigning IP addresses and usernames will be done by order of access, that is, the first user that accesses this service will be assigned the first IP address on the list.

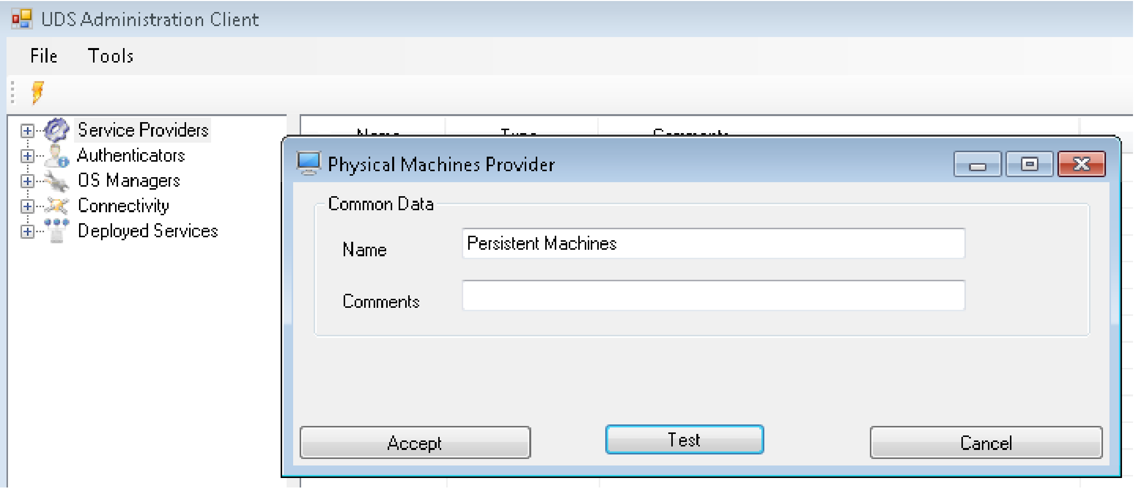
In order to connect to the machine to which the assigned IP address to a user belongs, the machine must have previously been switched on, the Terminal Services for Windows machines must be enabled and the NX software for Linux machines must be installed.

5.1.3.1 Register services provider "Physical Machines Providers"

Select "Physical Machines Provider".



For a "Physical Machines Provider," we must enter a name.


UDS Administration Client

Physical Machines Provider

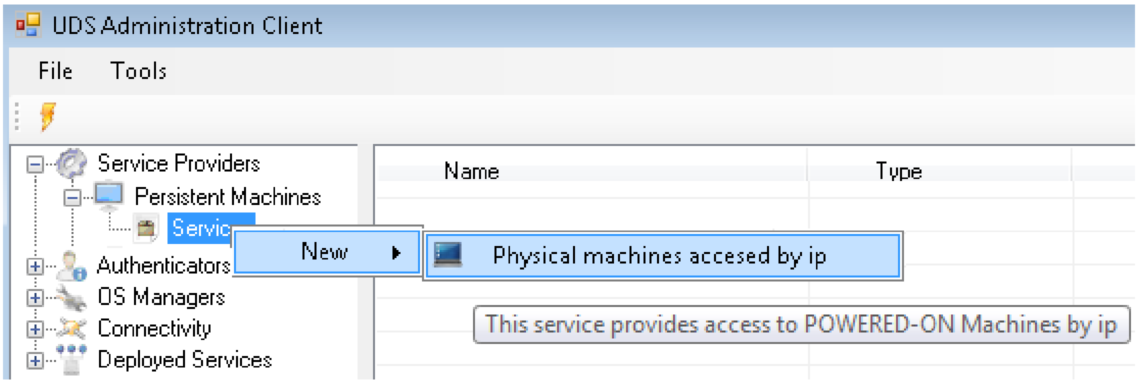
Common Data

Name: Persistent Machines

Comments:

5.1.3.2 Configuring service based on "Physical Machines Providers"

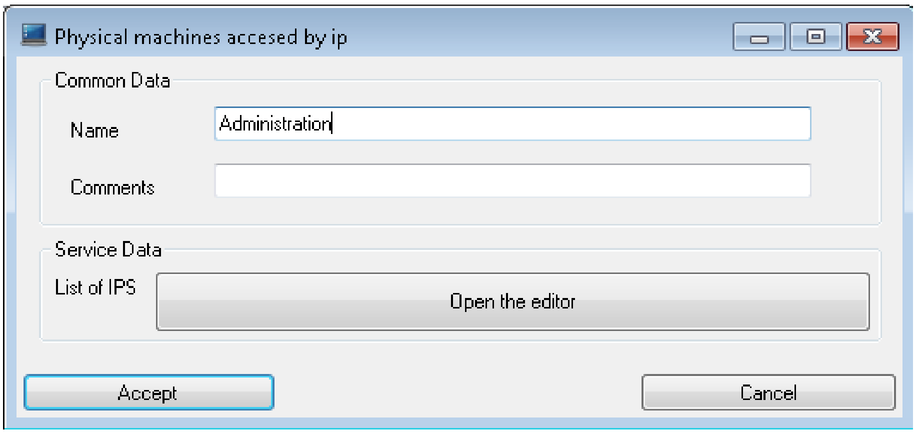
Once the Service Provider for persistent hardware has been created, you must register services based on "Physical machines accessed by IP."


UDS Administration Client

This service provides access to POWERED-ON Machines by ip

Choose a name for the service and enter the IP addresses to which they will provide access.

Click on the “Open the editor” button in the “Service Data List of IPS” section.



Physical machines accesed by ip

Common Data

Name: Administration

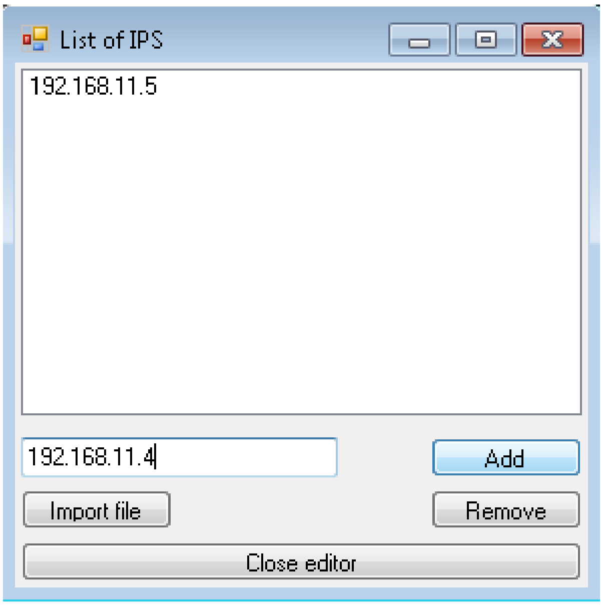
Comments:

Service Data

List of IPS

Enter the IP addresses of the machines to which we will have access and close the editor.

If we have a very long list of IP addresses, you can import the IP addresses from a text file by clicking on the “Import file” button.



List of IPS

192.168.11.5

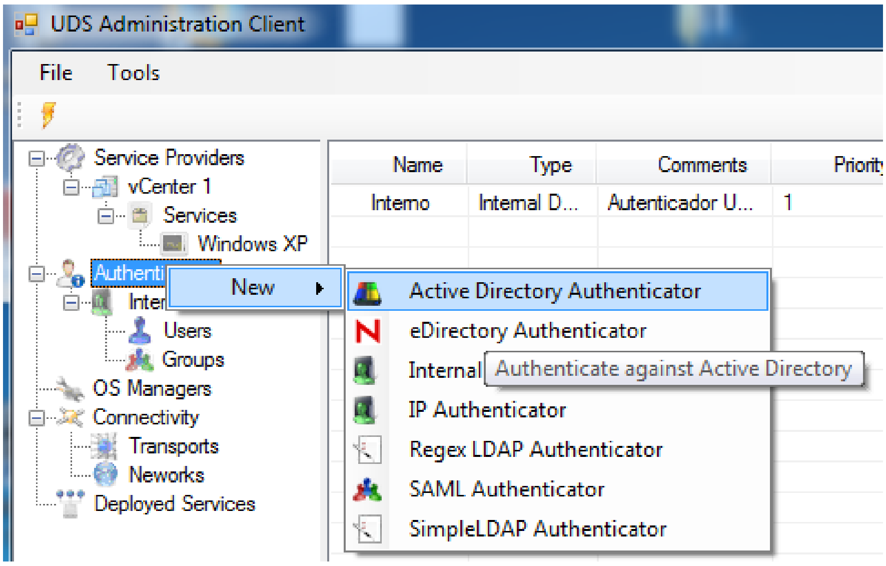
192.168.11.4

5.2 Configuring Authenticators

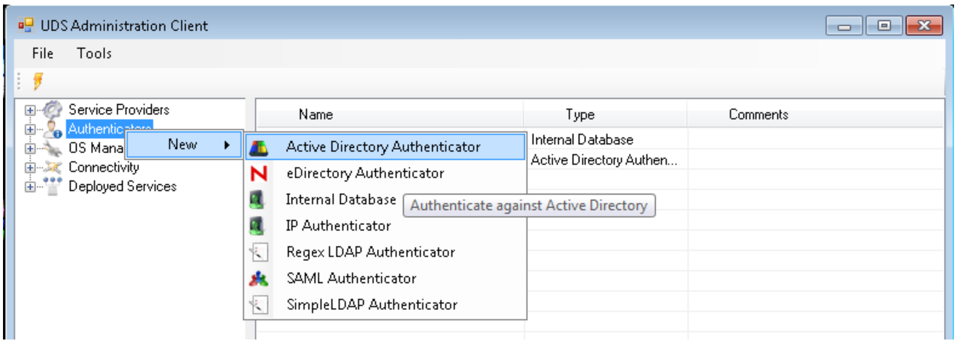
An Authenticator is a basic component within a desktop platform since it allows the users and user groups to whom you can grant sign in credentials to connect to the different virtual desktops.

An Authenticator is not a necessary component to create a "Deployed Services", but if it hasn't at least one assigned, there won't be users that will be able to make connections with virtual desktops in UDS platform.

You can choose between different types of authenticators.



5.2.1 Active Directory Authenticator



In an Active Directory Authenticator, we configure the authenticator name, the domain controller IP (“Host” field), a user (“Ldap User” field) and password with reading rights on the Active Directory.

The username (“Ldap User” field) must be typed in with the format *user@domain.*

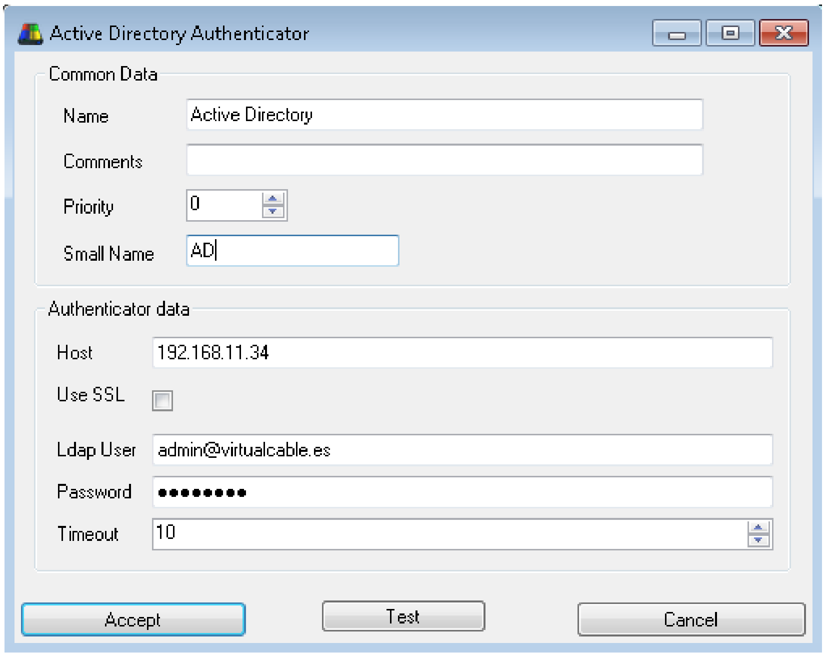
We can also indicate: the priority of this authenticator, the lower that priority is, the higher it will appear on the list of authenticators available in the user access window (this field admits negative values). We can also choose if we want to use an SSL connection and the Timeout in the connection with the Active Directory.

If the “Small Name” field is chosen, it only allows this “authenticator” to appear on the user login screen; to do this, we have to access the login screen with the following format:

address_server_broker/Small_Name

For example: *https://BrokerVC/AD*

By clicking on the "Test" button, we can check to see whether the connection has been made correctly.



Common Data

Name: Active Directory

Comments:

Priority: 0

Small Name: AD

Authenticator data

Host: 192.168.11.34

Use SSL: □

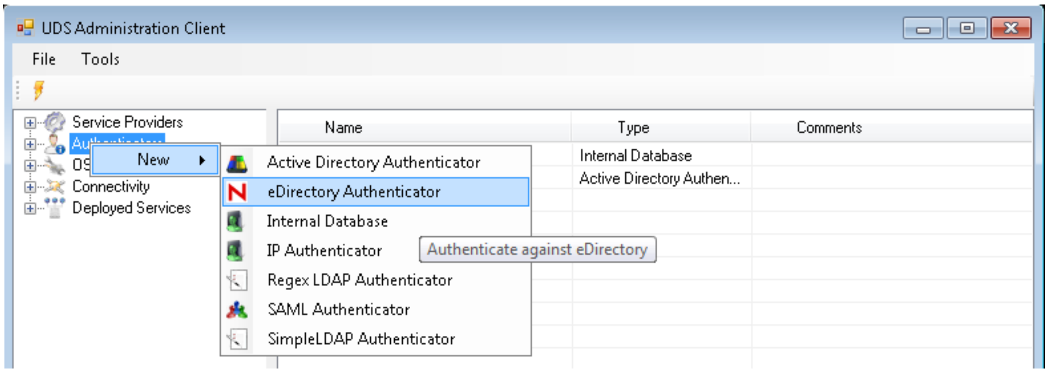
Ldap User: admin@virtualcable.es

Password: ●●●●●●●●●

Timeout: 10

5.2.2 eDirectory Authenticator

This authenticator is available to provide Novell network users and user groups access to UDS virtual desktops.



Name	Type	Comments
	Internal Database	
	Active Directory Authen...	

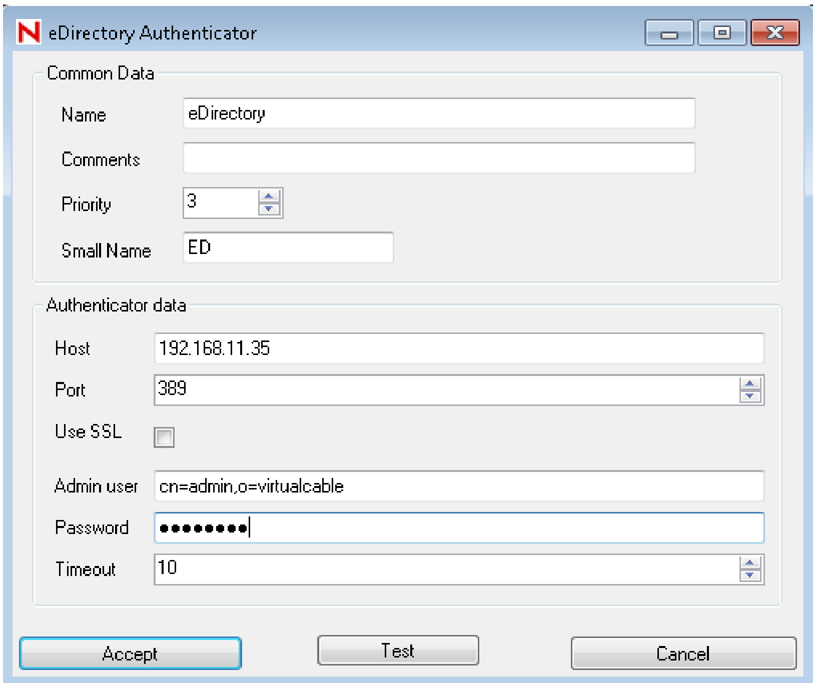
In an eDirectory Authenticator, we configure the authenticator name, the eDirectory IP server (Host field), a username (Admin User field) and password with reading rights on the eDirectory.

We can also indicate: the priority of this authenticator, the lower that priority is, the higher it will appear on the list of authenticators available in the user access window (this field admits negative values). We can also choose if we want to use an SSL connection and the Timeout in the connection with the Active Directory.

If the "Small Name" field is chosen, it only allows this "authenticator" to appear on the user login screen; to do this, we have to access the login screen with the following format: address_server_broker/Small_Name

For example: https://BrokerVC/ED

By clicking on the "Test" button, we can check to see whether the connection has been made correctly.



eDirectory Authenticator

Common Data

Name: eDirectory

Comments:

Priority: 3

Small Name: ED

Authenticator data

Host: 192.168.11.35

Port: 389

Use SSL: □

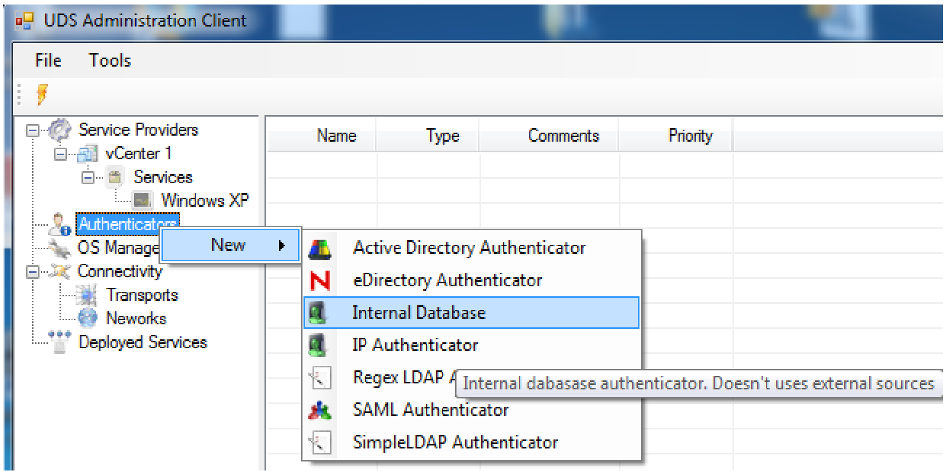
Admin user: cn=admin,o=virtualcable

Password: ●●●●●●●●

Timeout: 10

5.2.3 Internal Database

In environments where no external authenticator is available, it is possible to use the internal authenticator. This authenticator is included in the UDS broker and permits you to manually create users and user groups so that they can subsequently access the different services provided by UDS.



Internal dabasase authenticator. Doesn't uses external sources

In an Internal Database, we configure the authenticator name.

We can also indicate the priority of this authenticator. The lower that priority is, the higher it will appear on the list of authenticators available in the user access window (this field admits negative values).

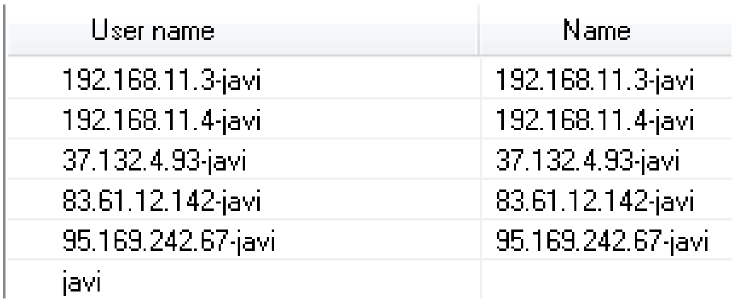
If the "Small Name" field is chosen, it only allows this "authenticator" to appear on the user login screen; to do this, we have to access the login screen with the following format: address_server_broker/Small_Name

For example: https://brokerVC/IN

In the section “Authenticator data”, we see two options:

“Different user for each host”: This option allows connections to virtual desktops by using a single connection user ID. These types of connections are made by creating multiple users in the internal database with the following username structure:

Hostname terminal ID + Connection User ID

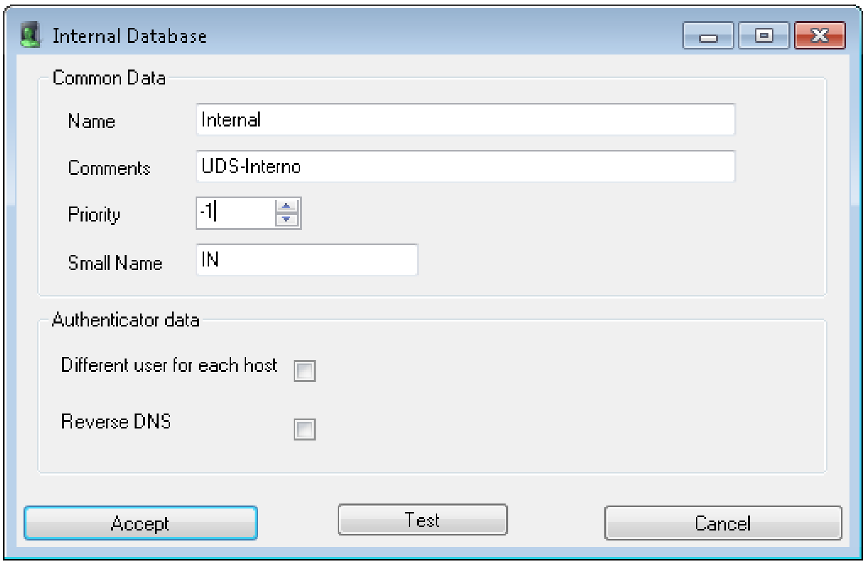


User name	Name
192.168.11.3-javi	192.168.11.3-javi
192.168.11.4-javi	192.168.11.4-javi
37.132.4.93-javi	37.132.4.93-javi
83.61.12.142-javi	83.61.12.142-javi
95.169.242.67-javi	95.169.242.67-javi
javi	

“Reverse DNS”: The behavior is exactly the same as in the previous option, but the username structure would be:

Hostname terminal ID + User ID

In order to be able to use this option, you must have the reverse DNS resolution in the connection terminal IDs. In the event that this does not exist, the username structure would continue using the IP address connection terminal ID.



Internal Database

Common Data

Name: Internal

Comments: UDS-Interno

Priority: -1

Small Name: IN

Authenticator data

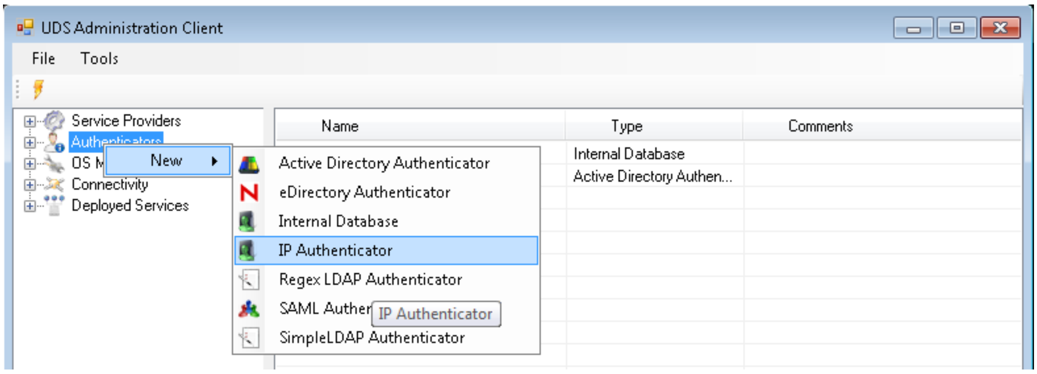
Different user for each host □

Reverse DNS □

5.2.4 IP Authenticator

It is possible to assign virtual desktops to connecting devices via the IP identifier.

(Ex: Thin Clients in kiosk mode, Call Center environments, proprietary applications, etc...)

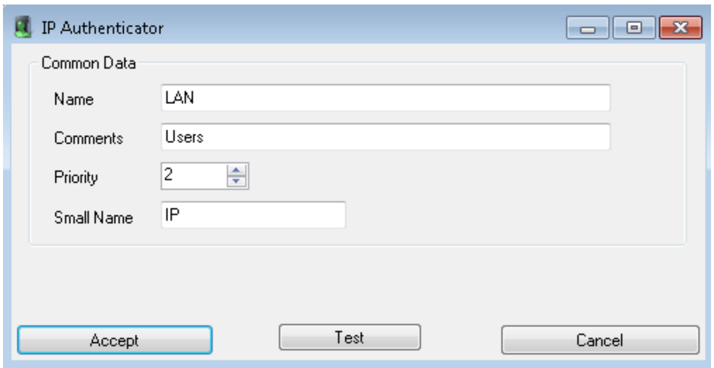


In an IP Authenticator, we configure the authenticator name.

We can also indicate the priority of this authenticator. The lower that priority is, the higher it will appear on the list of authenticators available in the user access window (this field admits negative values).

If the "Small Name" field is chosen, it only allows this "authenticator" to appear on the user login screen. To do this, we have to access the login screen with the following format: address_server_broker/Small_Name

For example: https://BrokerVC/IP



Name: LAN

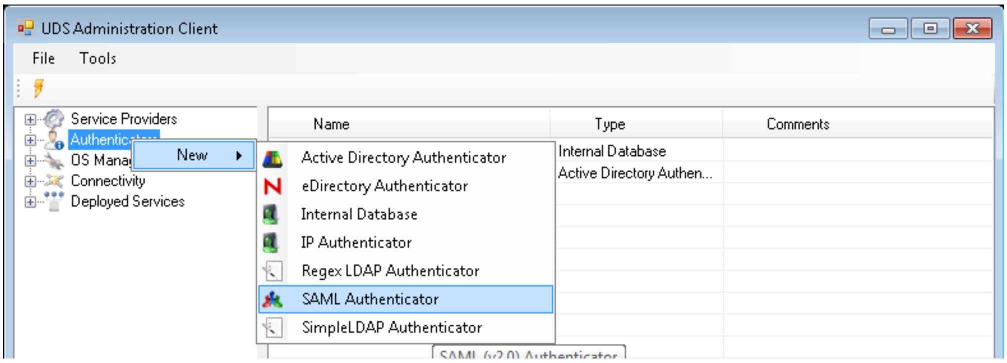
Comments: Users

Priority: 2

Small Name: IP

5.2.5 SAML Authenticator

SAML is used to exchange authentication and authorization data between security domains, that is, between an identity provider (an assertion producer) and a service provider (an assertion consumer).

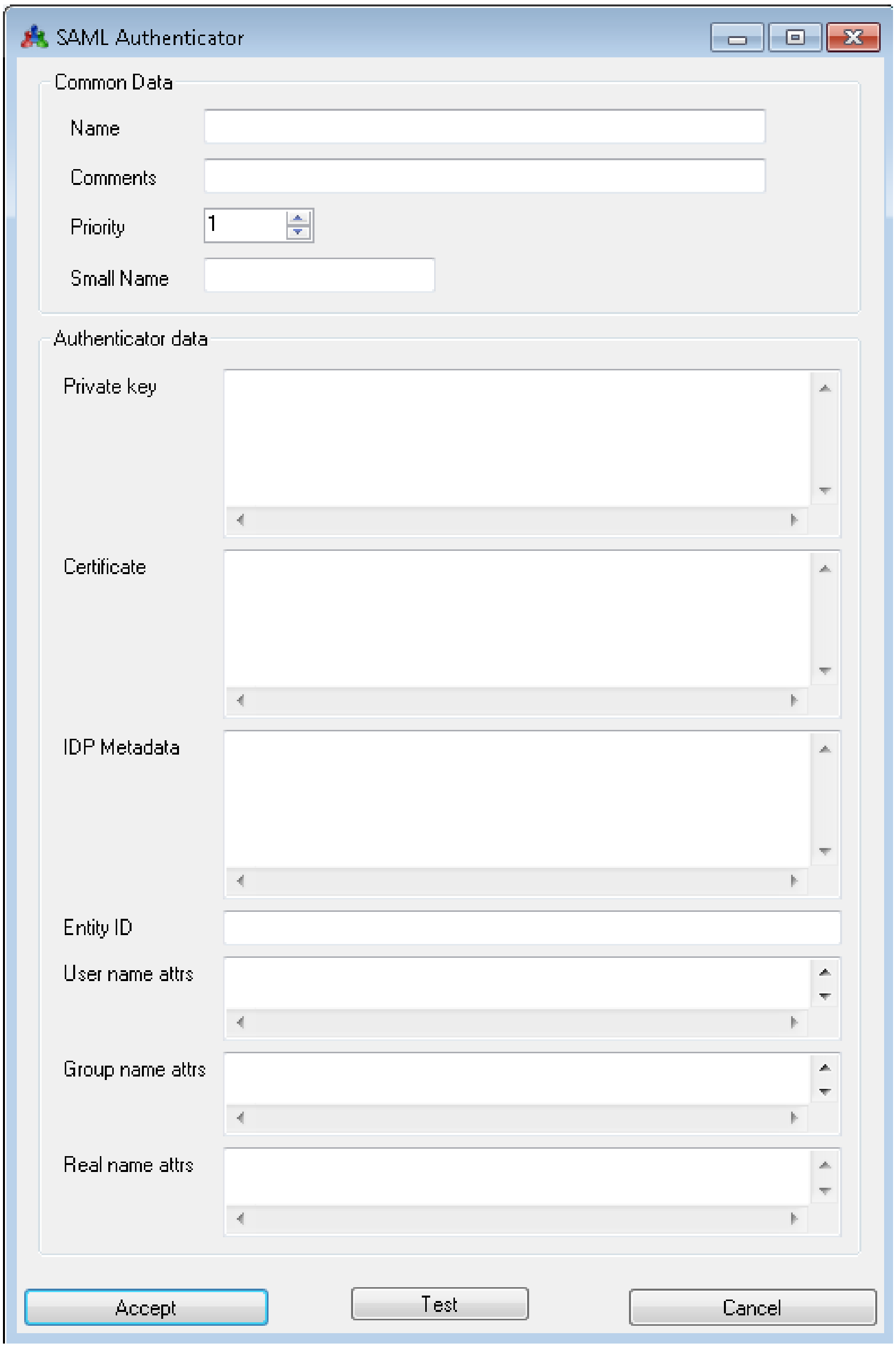


In a SAML Authenticator, we configure the authenticator name and data: Private Key, Certificate IDP Metedata, Entity ID, User name attrs, Group name attrs and Real name attrs.

We can also indicate the priority of this authenticator. The lower that priority is, the higher it will appear on the list of authenticators available in the user access window (this field admits negative values).

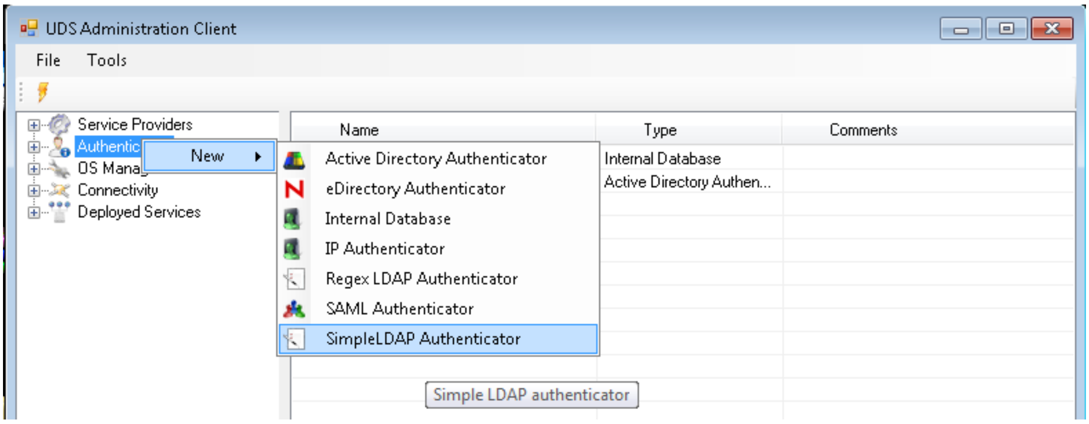
If the “Small Name” field is chosen, it only allows this “authenticator” to appear on the user login screen. To do this, we have to access the login screen with the following format: address_server_broker/Small_Name

For example: https://BrokerVC/SL


SAML Authenticator
Common Data
Name
Comments
Priority
1
Small Name
Authenticator data
Private key
Certificate
IDP Metadata
Entity ID
User name attrs
Group name attrs
Real name attrs
Accept
Test
Cancel

5.2.6 LDAP Authenticator

This is a generic authenticator available within the UDS platform. By configuring the correct parameters according to each case, we can define practically any authentication service based on LDAP.



In an LDAP Authenticator (Simple or Regex), we configure the authenticator name, the LDAP server IP (“Host” field), the connection port, a username (“Ldap User” field) and password with reading rights on LDAP, the name of the user and groups search base (“base” field) in the format: *dc=nombre_dominio,dc=extensión_dominio*).

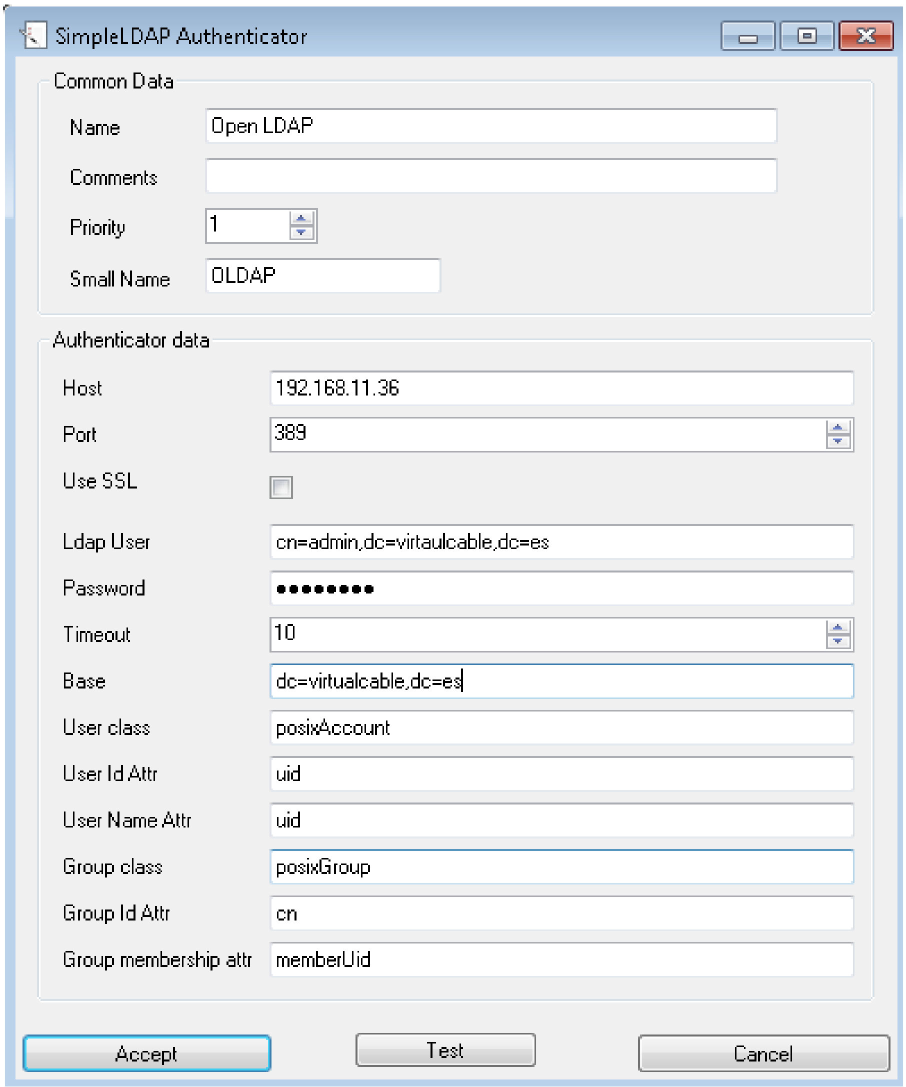
The username (Ldap User field) must be typed in with the format: *cn=user,dc=name_domain,dc=extension_domain*

We can also indicate the priority of this authenticator. The lower that priority is, the higher it will appear on the list of authenticators available in the user access window (this field admits negative values). We can also choose if we want to use an SSL connection and the Timeout in the connection with the LDAP Server.

If the “Small Name” field is chosen, it only allows this “authenticator” to appear on the user login screen; to do this, we have to access the login screen with the following format:

address_server_broker/Small_Name

For example: *https://BrokerVC/OLDAP*



SimpleLDAP Authenticator

Common Data

Name	Open LDAP
Comments	
Priority	1
Small Name	OLDAP

Authenticator data

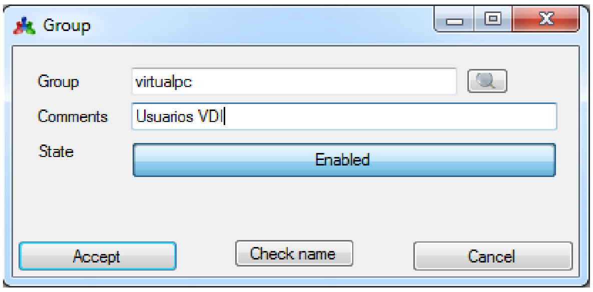
Host	192.168.11.36
Port	389
Use SSL	□
Ldap User	cn=admin,dc=virtaulcable,dc=es
Password	●●●●●●●●●
Timeout	10
Base	dc=virtualcable,dc=es
User class	posixAccount
User Id Attr	uid
User Name Attr	uid
Group class	posixGroup
Group Id Attr	cn
Group membership attr	memberUid

5.3 Configuring users, groups and users metagroups

Once the authenticator or authenticators have been configured, you must configure the user groups that contain the users to whom access to the virtual desktops is to be granted. It is also possible to create metagroups, which will be used to combine several groups.

The groups, metagroups and users can be temporarily activated or deactivated.

Searching for user groups is done automatically in all of the defined authenticators in UDS, with the exception of “Internal” and “by IP” authenticators, in which the groups are registered without being able to perform a search. Therefore, it is important to ensure that the group name provided is correct.



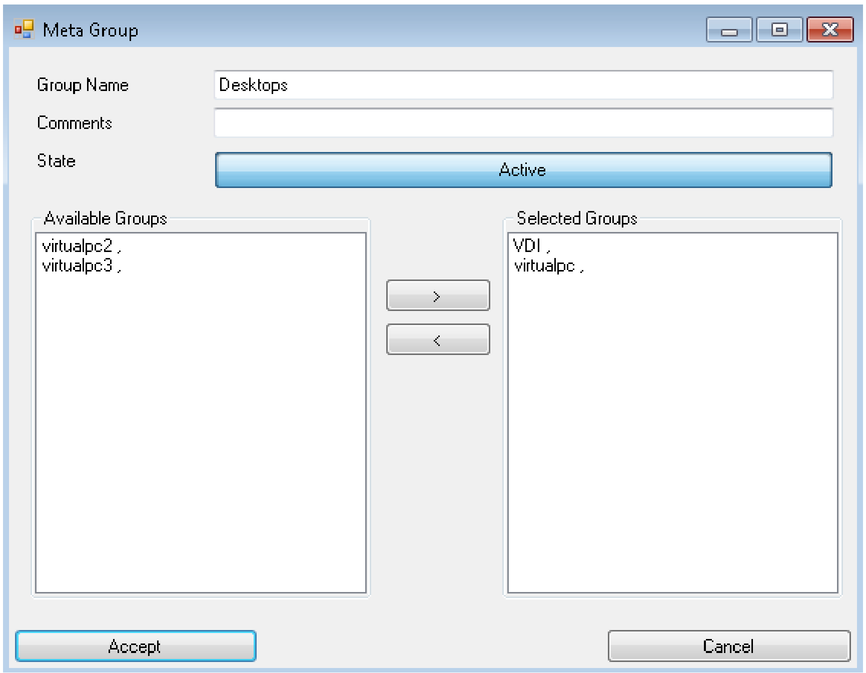
Group

Group: virtualpc

Comments: Usuarios VDI

State: Enabled

To create a metagroup, we select the groups that will form part of the metagroup, we choose a name for the new group and we click "Accept."



Group Name: Desktops

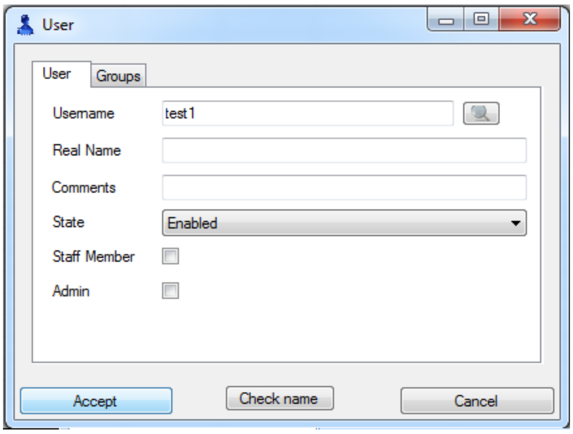
Comments:

State: Active

Available Groups:
virtualpc2 ,
virtualpc3 ,

Selected Groups:
VDI ,
virtualpc ,

The users of the configured groups are automatically added to the system when they connect to the UDS platform for the first time, except in "Internal" or "by IP" authenticators, in which the users will have to be manually registered.



Username: test1

Real Name:

Comments:

State: Enabled

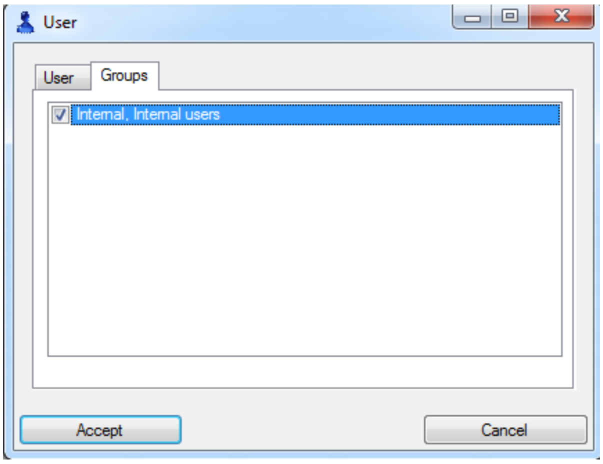
Staff Member □

Admin □

The additional "Staff" parameter allows access to downloads (UDS actor) and to the UDS administration.

The additional "Admin" parameter allows access to downloads (UDS actor), to the administration and also allows for the modification of advanced UDS configurations (Tab "Tools" - "Configuration"). An "Admin" user has to simultaneously be a "Staff" member.

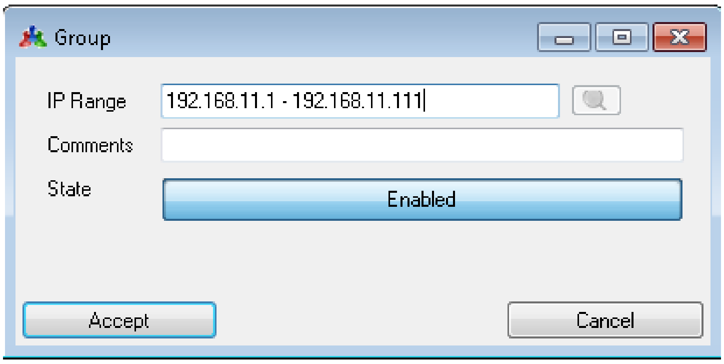
After creating a user in the "Internal" and "by IP" authenticators, the user must be assigned to a group.



☑ Internal, Internal users

The creation of an authenticator group "by IP" is different from the rest, given that in this case a range of IP addresses will be registered in order to provide access to all of the equipment within this range. This range of addresses is defined in the following way:

IP address start range – IP address end range



IP Range: 192.168.11.1 - 192.168.11.111

Comments:

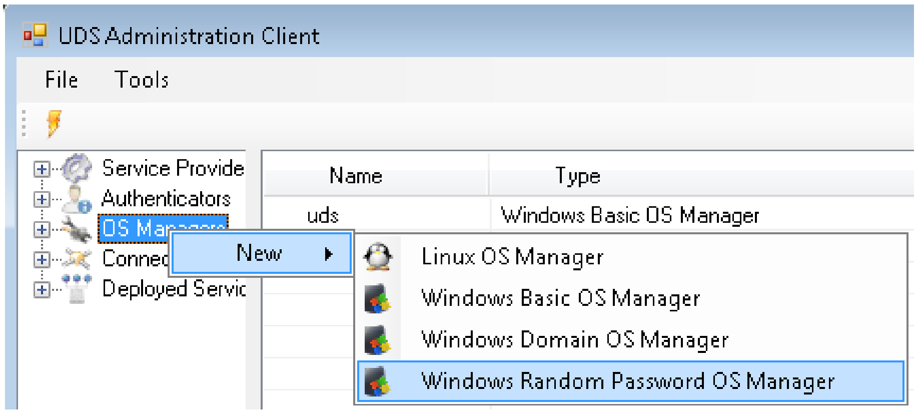
State: Enabled

5.4 Configuring “OS Managers”

An OS Manager initiates a previously configured type of service.

The UDS Actor, hosted on the virtual machine, is responsible for the interaction between the OS and the broker based on the configurations or type of OS Manager chosen.

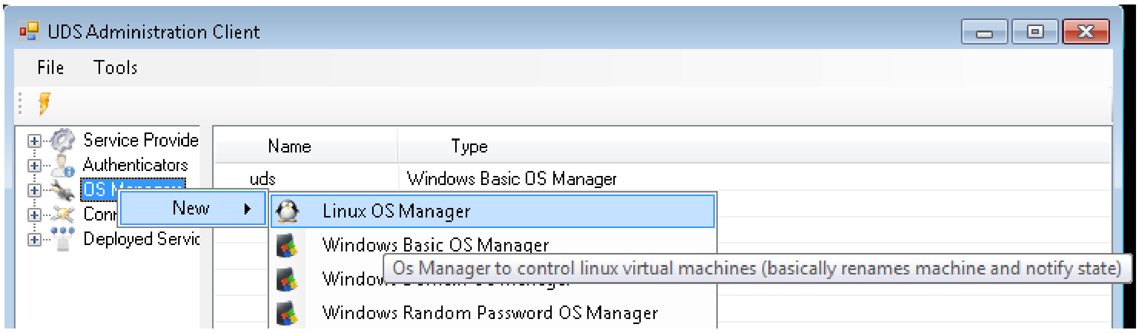
In order to perform VDI deployments via Linked Clones, you will have to select the disconnection behavior of the linked clones, within the configuration of each OS Manager.



Name	Type
uds	Windows Basic OS Manager

5.4.1 Linux OS Manager

A "Linux OS Manager" is used for virtual desktops based on Linux systems.



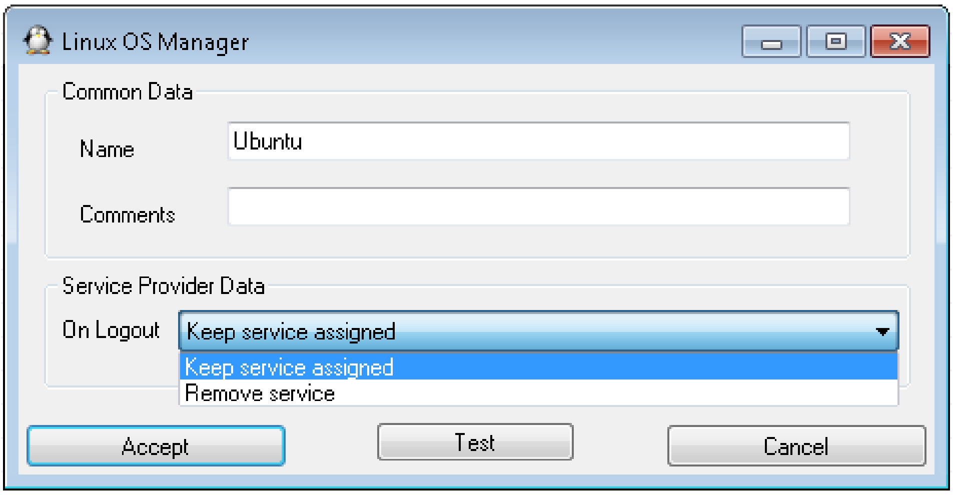
Name	Type
uds	Windows Basic OS Manager

Os Manager to control linux virtual machines (basically renames machine and notify state)

In order to configure Linux OS Manager, enter the name and configure which action the system will perform when a user disconnects:

- Keep service assigned: When a user logs out the desktop won't undergo any change. If this same user requests again a virtual machine to the system, the system will provide the same virtual desktop.

- Remove service: When a user logs out, the system will destroy the desktop. If this same user requests again a virtual machine to the system, the system will provide a new virtual desktop.



Linux OS Manager

Common Data

Name: Ubuntu

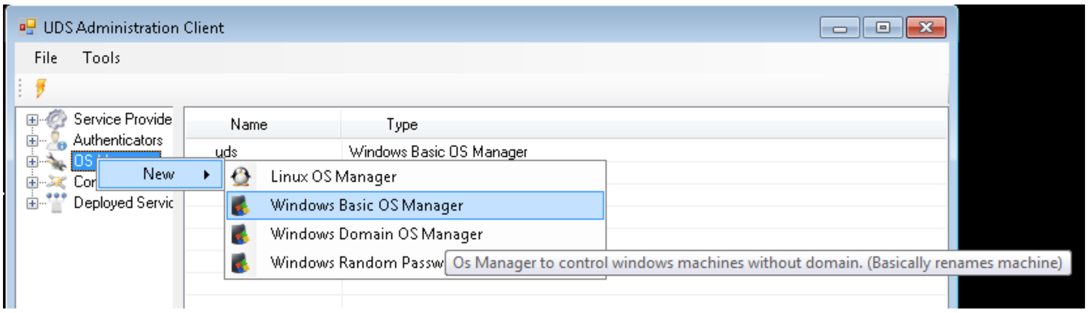
Comments:

Service Provider Data

On Logout: Keep service assigned

5.4.2 Windows Basic OS Manager

A "Windows Basic OS Manager" is used for virtual desktops based on Windows systems which are not part of a domain.

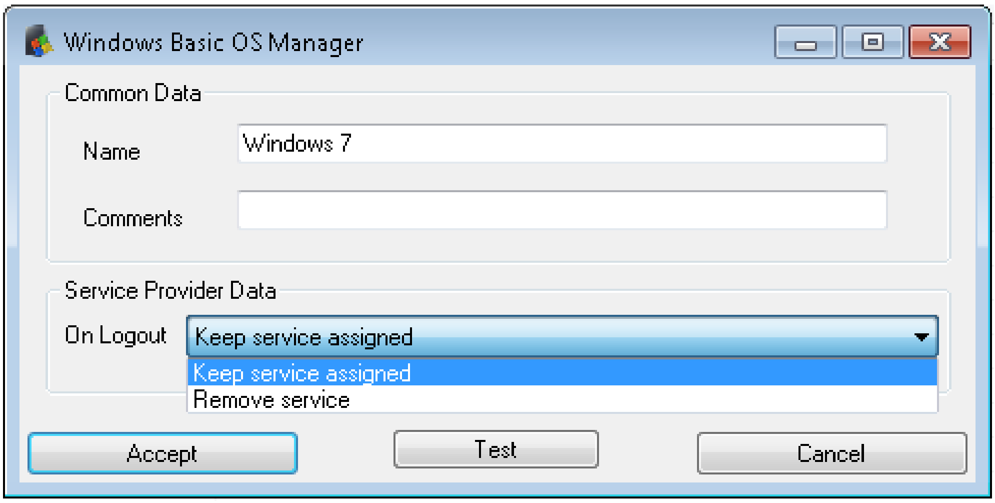


Os Manager to control windows machines without domain. (Basically renames machine)

In order to configure a Windows Basic OS Manager, enter the name and configure which action the system will perform when a user disconnects.

- Keep service assigned: When a user logs out the desktop won't undergo any change. If this same user requests again a virtual machine to the system, the system will provide the same virtual desktop.

- Remove service: When a user logs out, the system will destroy the desktop. If this same user requests again a virtual machine to the system, the system will provide a new virtual desktop.



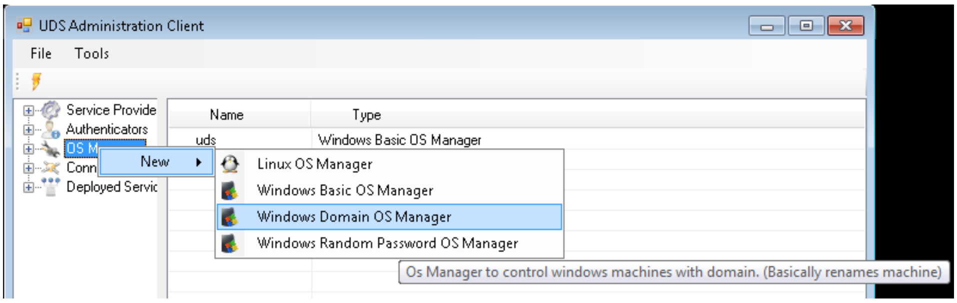
Name: Windows 7

Comments:

On Logout: Keep service assigned

5.4.3 Windows Domain OS Manager

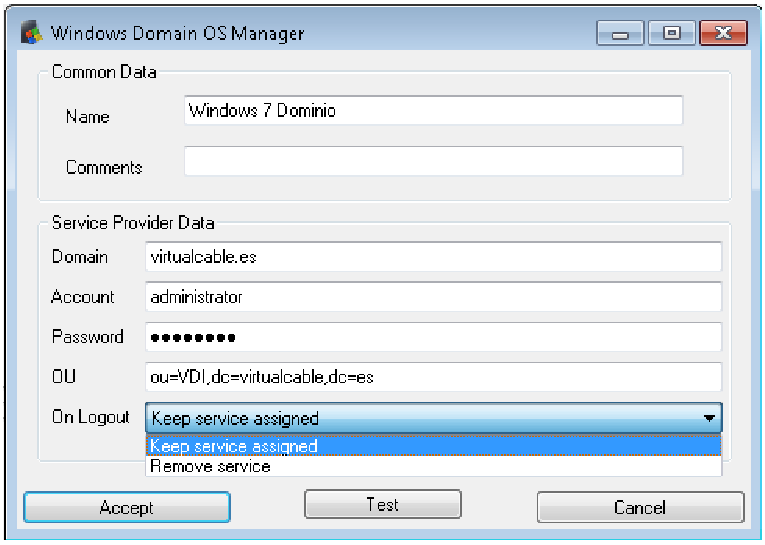
A "Windows Domain OS Manager" is used for virtual desktops based on Windows systems which are part of a domain.



Os Manager to control windows machines with domain. (Basically renames machine)

In order to configure a "Windows Domain OS Manager", enter the following data:

- The name of the OS Manager.
- The domain name to which the virtual desktops deployed with this OS Manager are going to belong.
- User credentials with permission to add machines to the domain.
- Information of the Organizing Unit (OU) where the virtual desktops deployed with this OS Manager are going to be registered (if we don't write anything, the desktops will be located in the branch by default).
- Configure the action the system will perform when a user logs out:
    - Keep service assigned: When a user logs out the desktop won't undergo any change. If this same user requests again a virtual machine to the system, the system will provide the same virtual desktop.
    - Remove service: When a user logs out, the system will destroy the desktop. If this same user requests again a virtual machine to the system, the system will provide a new virtual desktop.


Windows Domain OS Manager

Common Data

Name: Windows 7 Dominio

Comments:

Service Provider Data

Domain: virtualcable.es

Account: administrator

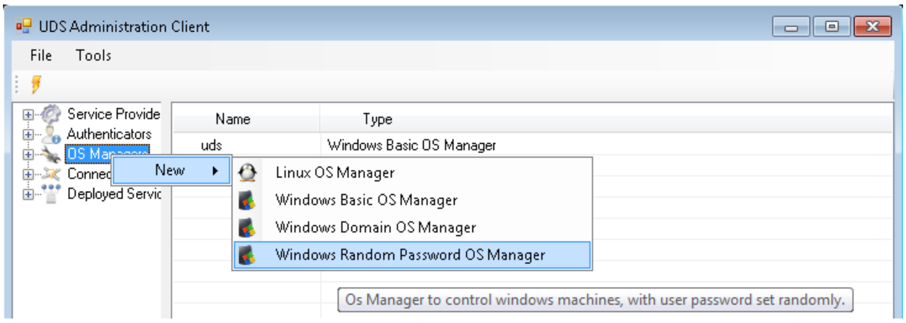
Password: ●●●●●●●●●

OU: ou=VDI,dc=virtualcable,dc=es

On Logout: Keep service assigned

5.4.4 Windows Random Password OS Manager

A "Windows Random Password OS Manager" is used for virtual desktops based on Windows systems that are not part of a domain and require a higher level of security in the user access.


UDS Administration Client

Name	Type
uds	Windows Basic OS Manager

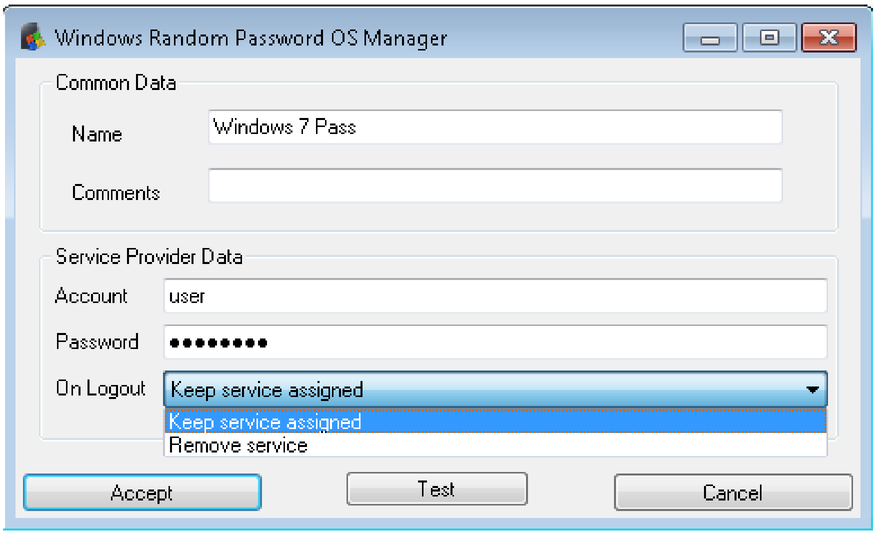
Os Manager to control windows machines, with user password set randomly.

Using this assigns a random password, previously defined during configuration, to an existing local user in each new deployed virtual desktop, thus providing a higher level of access security.

To configure a "Windows Random Password OS Manager" enter the following data:

- OS Manager Name.
- Local user defined in the template.
- Password initially established for the local user in the template.

- Configure which action the system will perform when a user disconnects:
    - Keep service assigned: When a user logs out the desktop won't undergo any change. If this same user requests again a virtual machine to the system, the system will provide the same virtual desktop.
    - Remove service: When a user logs out, the system will destroy the desktop. If this same user requests again a virtual machine to the system, the system will provide a new virtual desktop.



Windows Random Password OS Manager

Common Data

Name: Windows 7 Pass

Comments:

Service Provider Data

Account: user

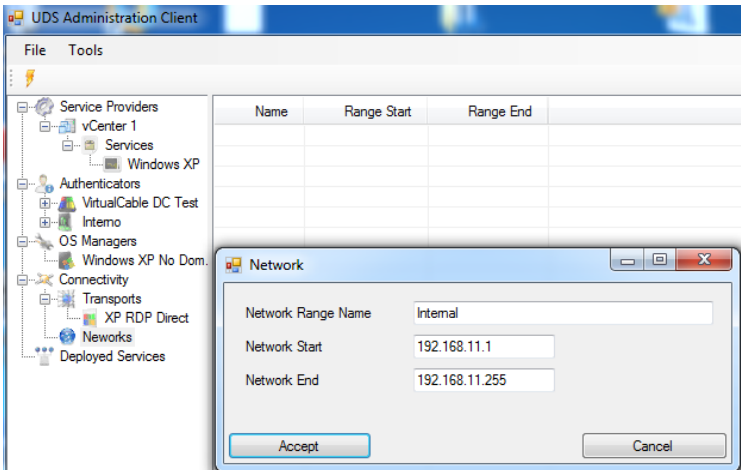
Password: ●●●●●●●●●

On Logout: Keep service assigned

5.5 Configuring “Networks”

UDS allows registering several networks to allow or deny access to virtual desktops. These networks, together with Transports will define what kind of access the users will have to their virtual desktops generated by UDS.

In "Connectivity" section we can configure a new access network indicating: "Network Range Name" (descriptive for each network), “Network Start” and “Network End”.



Network Range Name: Internal
Network Start: 192.168.11.1
Network End: 192.168.11.255

If no network is registered, access to the virtual desktops will be allowed from any network.

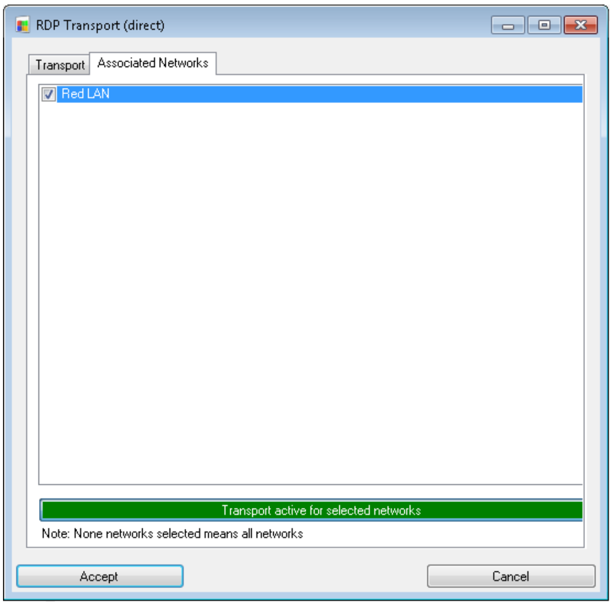
5.6 Configuring “Transports”

In order to connect to the virtual desktops, you must create Transports. These are small applications that will be run on the client and which will be responsible for providing access to the implemented service.

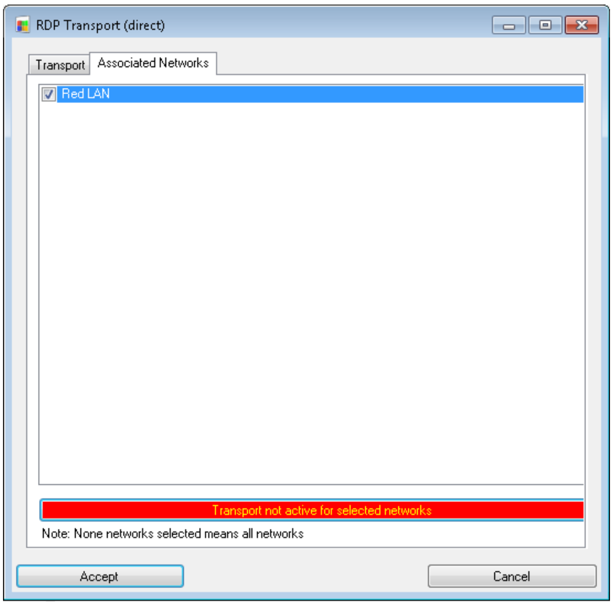
In “Associated Networks” tab, inside any “Transports”, the available networks to make the connection to the virtual desktop through the selected transport are defined.

A single transport can use different networks, as needed

In the event that a network is selected and the transport is marked as "active" for the selected network (in green), the connection will be made with the active transport via the network.


RDP Transport (direct)
Transport
Associated Networks
☑ Red LAN
Transport active for selected networks
Note: None networks selected means all networks
Accept
Cancel

In the event that a network is selected and the transport is marked as "inactive" for the selected network (in red), the connection will not be allowed for the transport selected via this network.


RDP Transport (direct)
Transport
Associated Networks
☑ Red LAN
Transport not active for selected networks
Note: None networks selected means all networks
Accept
Cancel

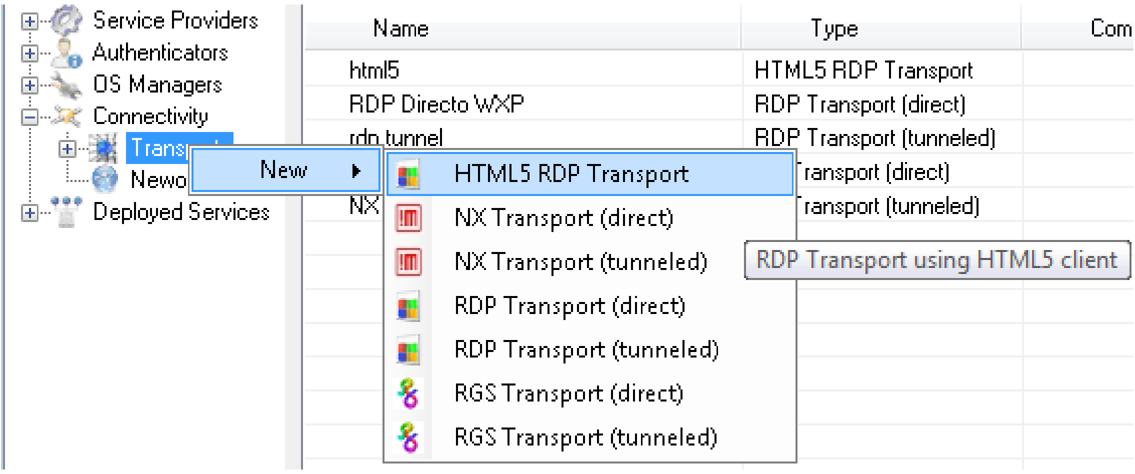
The “Associated Networks” tab will appear empty until the different networks are configured.

The following transports are defined in UDS:

5.6.1 HTML5 RDP Transport

A "HTML5 RDP Transport" allows Access to Windows and Linux virtual desktops through RDP protocol using a browser which supports HTML5 (for Linux desktops the machines must have the XRDP packet installed. For Windows desktops the RDP access need to be set up).

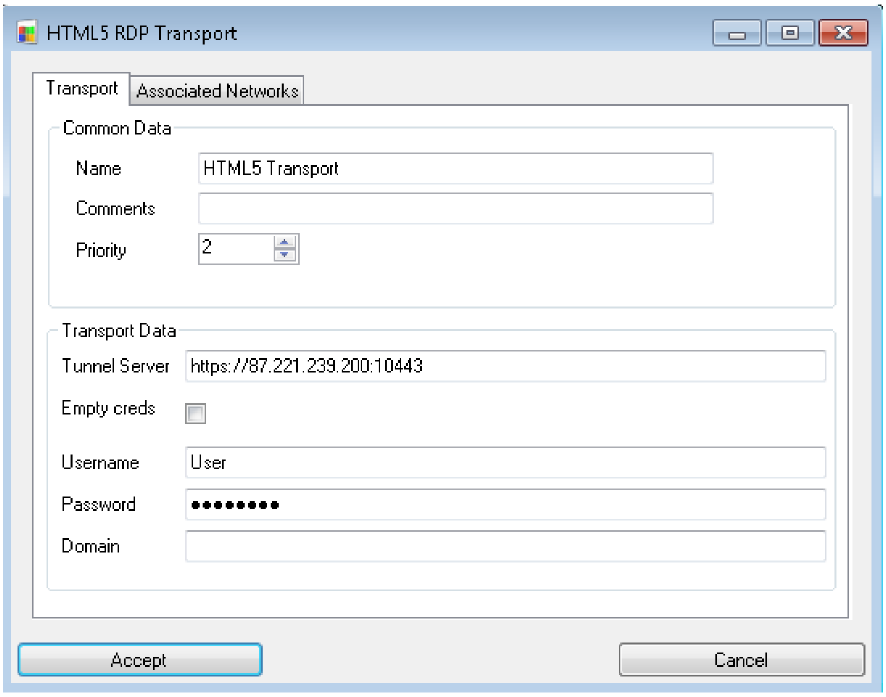
This transport uses UDS Tunneler server to make the connection against the virtual desktops. It has to be configured beforehand in order to work properly.



Name	Type	Com
html5	HTML5 RDP Transport	
RDP Directo WXP	RDP Transport (direct)	
rdp tunnel	RDP Transport (tunneled)	
	ransport (direct)	
NX	ransport (tunneled)	

For HTML5 RDP Transport, we configure the name of the transport, the IP address of the UDS tunneler server and port (“Tunnel Server” field) with the format: https://IP_Tunneler:10443 (port by default). We indicate whether we want to redirect specific credentials to the virtual desktop, and if the “Empty creds” box is checked, no credential will pass through to the virtual desktop. If we do not check the “Empty creds” box and enter a “username” and “password” (it is also possible to enter a domain name), these credentials will be redirected to the virtual desktops.

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values).



Name: HTML5 Transport

Comments:

Priority: 2

Tunnel Server: https://87.221.239.200:10443

Empty creds ☐

Username: User

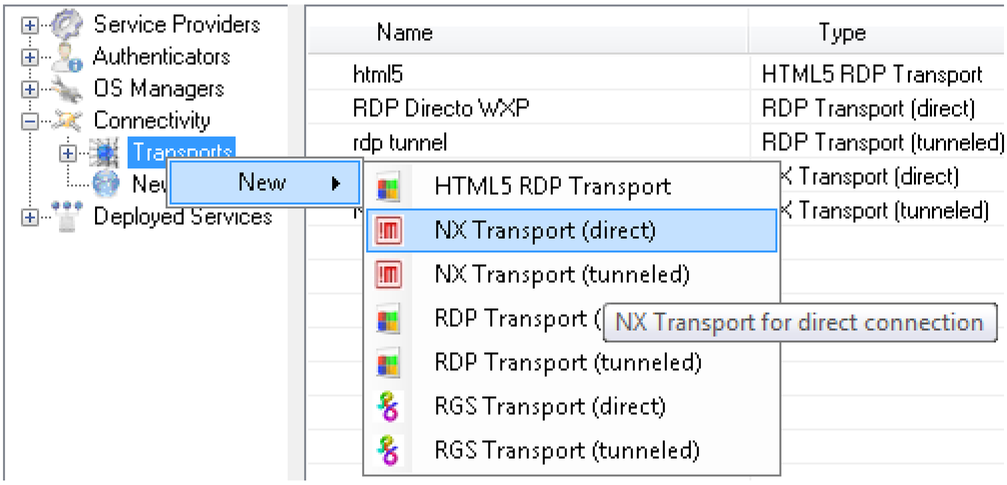
Password: ●●●●●●●●●

Domain:

5.6.2 NX Transport (direct)

A "NX Transport (direct)" allows Access to Linux virtual desktops through NX software (the virtual machines and the connection clients must have NX installed).

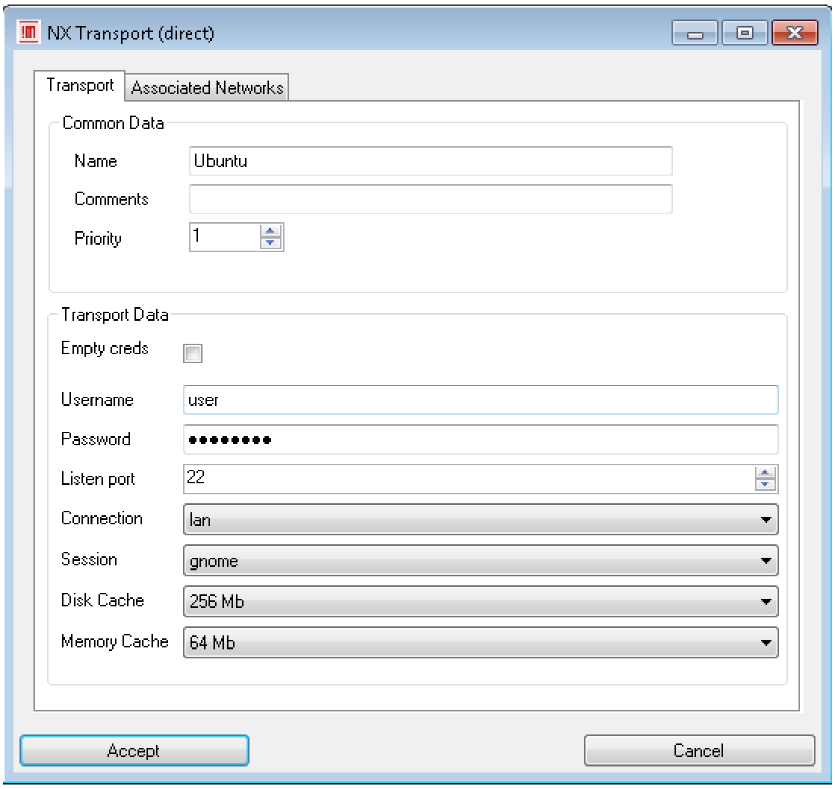
Currently, the NX supported version is 3.5



Name	Type
html5	HTML5 RDP Transport
RDP Directo WXP	RDP Transport (direct)
rdp tunnel	RDP Transport (tunneled)
	X Transport (direct)
	X Transport (tunneled)

For a NX Transport (direct), we configure the transport name, we indicate whether we want to redirect specific credentials to the virtual desktop. If the "Empty creds" box is checked, no credential will pass through to the virtual desktop. If we do not check the "Empty creds" box and enter a "username" and "password", these credentials will be redirected to the virtual desktops.

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values). We can also indicate what port and optimization connection parameters we want to use for the connection, such as: "Connection", "Session", "Disk Cache" and "Memory Cache".



Common Data

Name: Ubuntu

Comments:

Priority: 1

Transport Data

Empty creds □

Username: user

Password: ●●●●●●●●

Listen port: 22

Connection: lan

Session: gnome

Disk Cache: 256 Mb

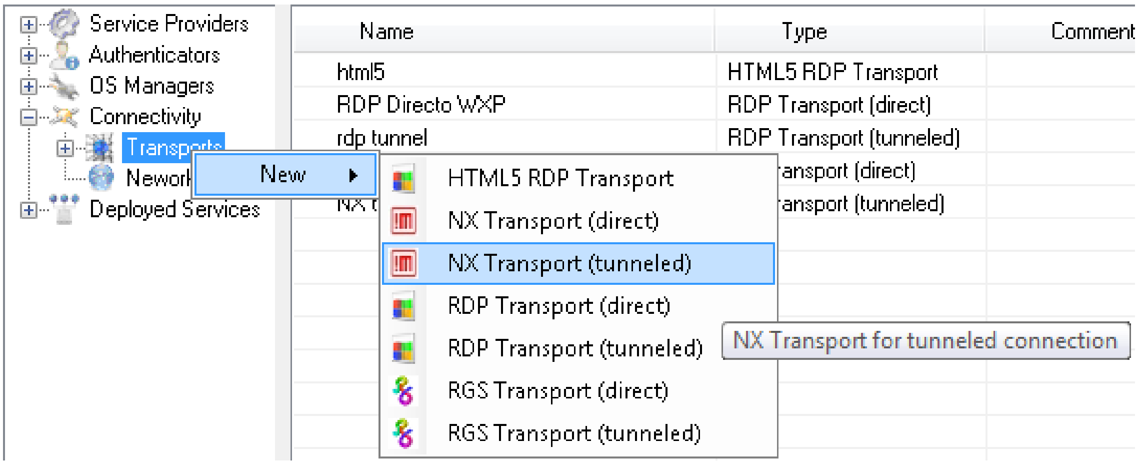
Memory Cache: 64 Mb

5.6.3 NX Transport (tunneled)

A "NX Transport (tunneled)" allows Access to Linux virtual desktops through NX software (the virtual machines and the connection clients must have NX installed).

Currently, the NX supported version is 3.5

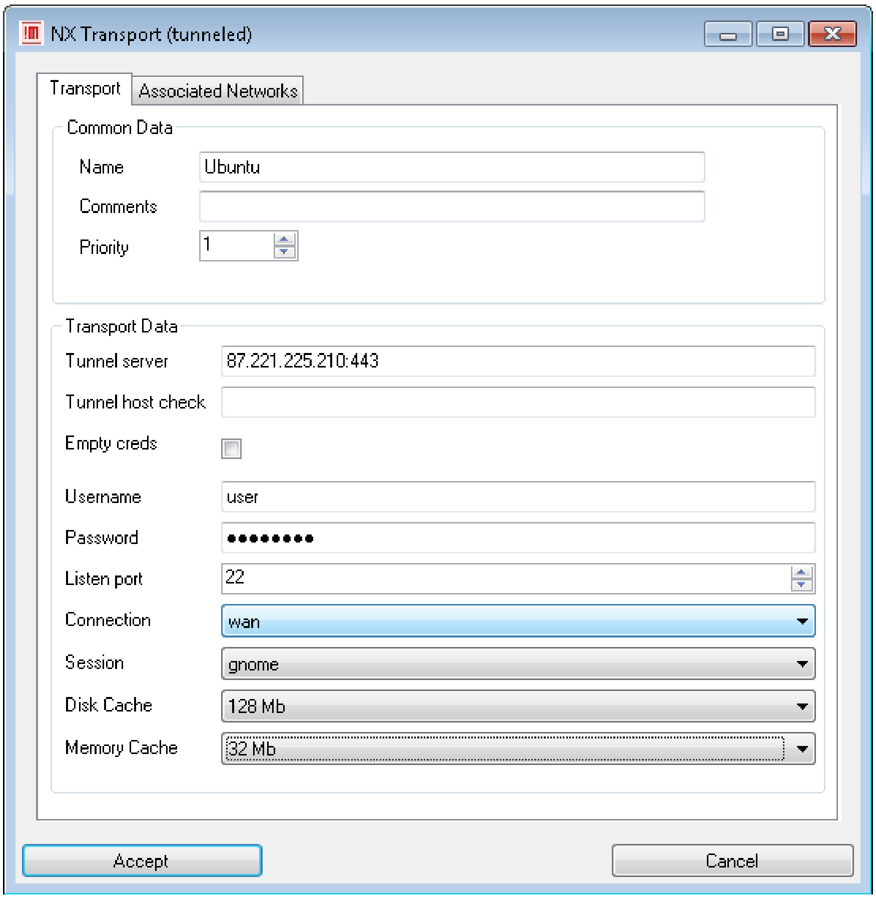
This transport uses UDS tunneler server to make the connection against the virtual desktops, and it needs to be configured beforehand in order to work properly.



Name	Type	Comment
html5	HTML5 RDP Transport	
RDP Directo WXP	RDP Transport (direct)	
rdp tunnel	RDP Transport (tunneled)	
	ansport (direct)	
NX (	ansport (tunneled)	

For a NX Transport (tunneled), we configure the transport name, the UDS Tunneler server IP address and a port ("tunnel server" field) with the format IP_Tunneler:443 (port by default). We indicate if we want to redirect specific credentials to the virtual desktop. If we check the "Empty creds" box, no credential will be passed to the virtual desktop. If we don't check the "Empty creds" box and we indicate a "username" and "password", these credentials will be redirected to the virtual desktop.

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values). We can also indicate what port and optimization connection parameters we want to use for the connection, such as: "Connection", "Session", "Disk Cache" and "Memory Cache".



Common Data

Name: Ubuntu

Comments:

Priority: 1

Transport Data

Tunnel server: 87.221.225.210:443

Tunnel host check:

Empty creds: □

Username: user

Password: ●●●●●●●●●

Listen port: 22

Connection: wan

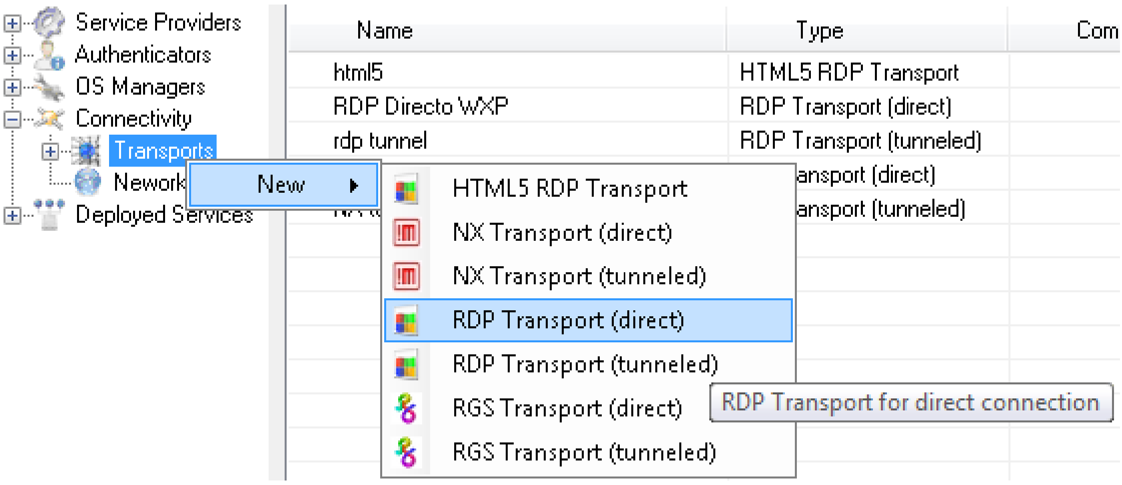
Session: gnome

Disk Cache: 128 Mb

Memory Cache: 32 Mb

5.6.4 RDP Transport (direct)

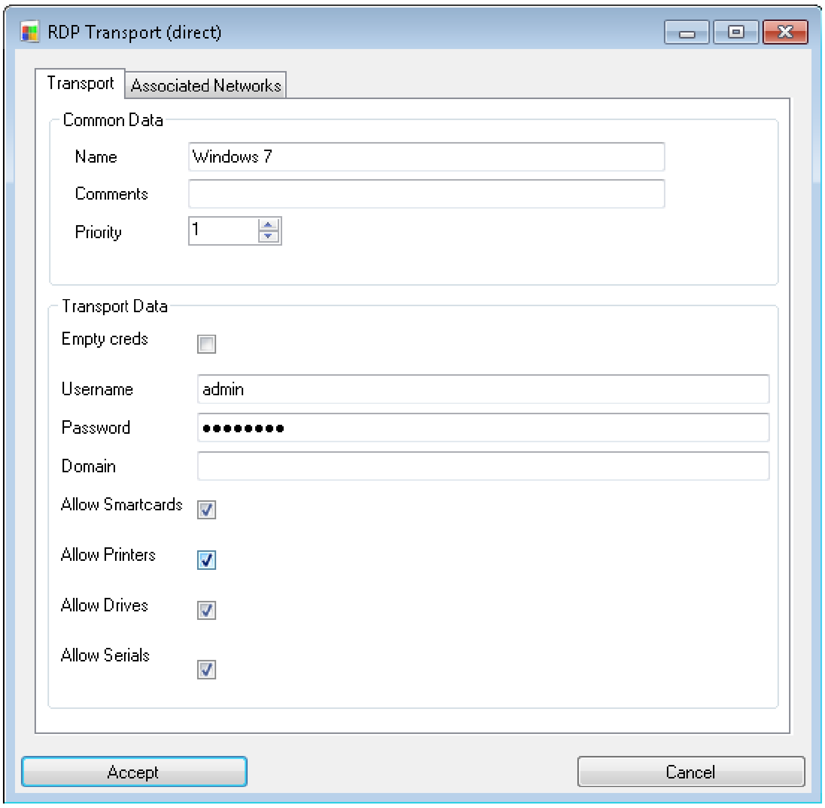
A "RDP Transport (direct)" allows access to Windows virtual desktops through RDP protocol (the virtual machines must have RDP service enabled).



Name	Type	Com
html5	HTML5 RDP Transport	
RDP Directo WXP	RDP Transport (direct)	
rdp tunnel	RDP Transport (tunneled)	
	ansport (direct)	
	ansport (tunneled)	

For the RDP Transport (direct), we configure the transport name, we indicate whether we want to redirect specific credentials to the virtual desktop. If the “Empty creds” box is checked, no credential will pass through to the virtual desktop. If we do not check the “Empty creds” box and enter a “username” and “password” (it is also possible to enter a domain name), these credentials will be redirected to the virtual desktop.

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values). We can also indicate the parameters "Allow Smartcards", "Allow Printers", "Allow Drives" and "Allow Serials".



Common Data

Name: Windows 7

Comments:

Priority: 1

Transport Data

Empty creds □

Username: admin

Password: ••••••••

Domain:

Allow Smartcards ☑

Allow Printers ☑

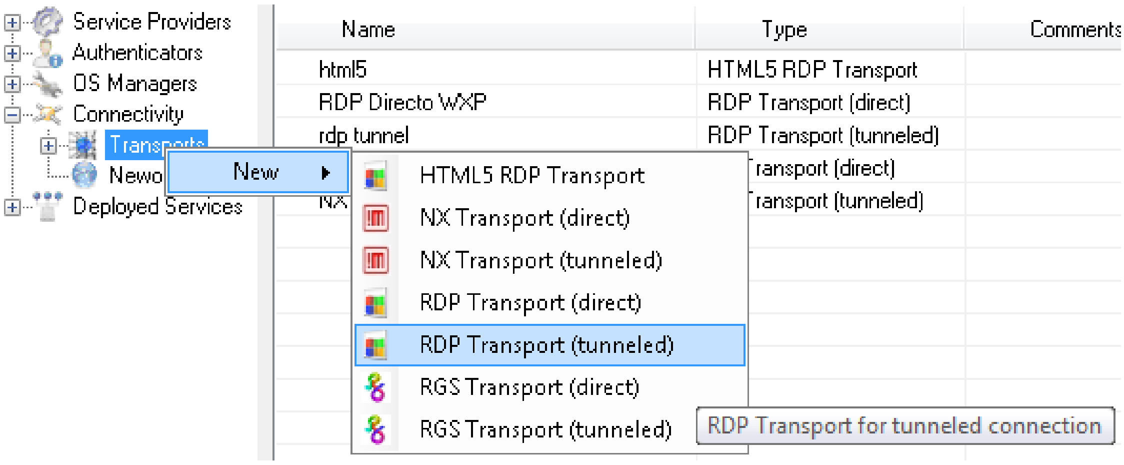
Allow Drives ☑

Allow Serials ☑

5.6.5 RDP Transport (tunneled)

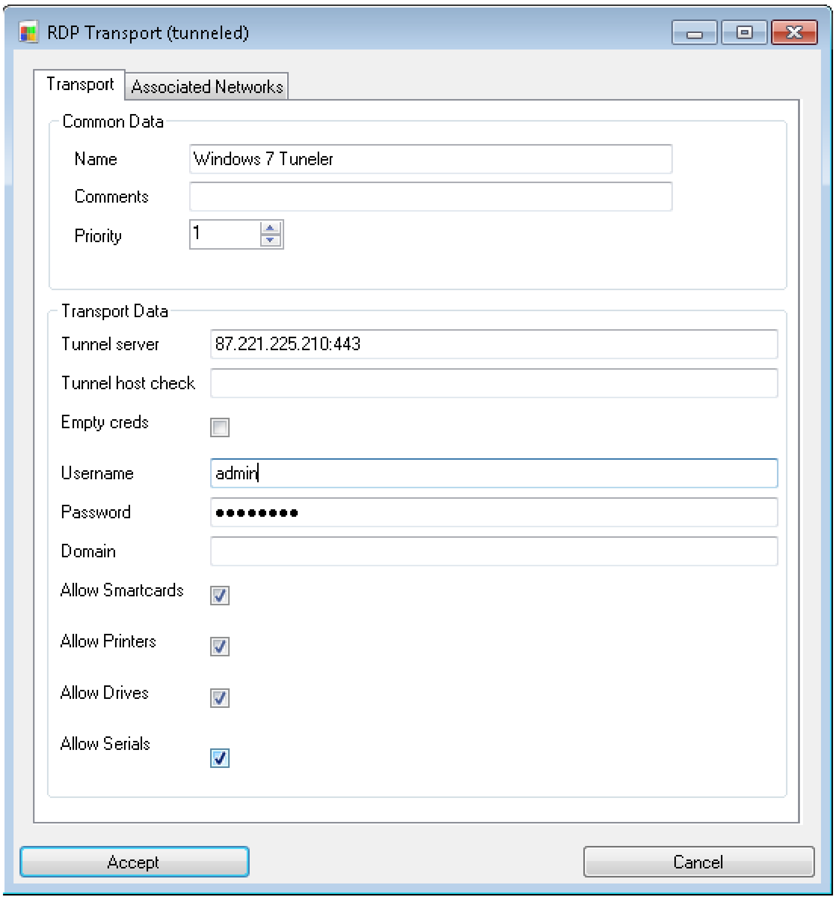
A "RDP Transport (tunneled)" allows access to Windows virtual desktops through RDP protocol (the virtual machines must have RDP service enabled).

This transport uses UDS Tunneler server to make the connection against the virtual desktops. It must be configured beforehand in order to work properly.



For the RDP transport (tunneled), we configure the transport name, the IP address of the UDS tunneler server and port (“Tunnel server” field) with the format: IP_Tunneler:443 (port by default). We indicate whether we want to redirect specific credentials to the virtual desktop, and if the “Empty creds” box is checked, no credential will pass through to the virtual desktop. If we do not check the “Empty creds” box and enter a “username” and “password” (it is also possible to enter a user domain), these credentials will be redirected to the virtual desktop.

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values). We can also indicate the parameters “Allow Smartcards”, “Allow Printers”, “Allow Drives” and “Allow Serials”.



Common Data

Name: Windows 7 Tuneler

Comments:

Priority: 1

Transport Data

Tunnel server: 87.221.225.210:443

Tunnel host check:

Empty creds: □

Username: admin

Password: ●●●●●●●●●

Domain:

Allow Smartcards: ☑

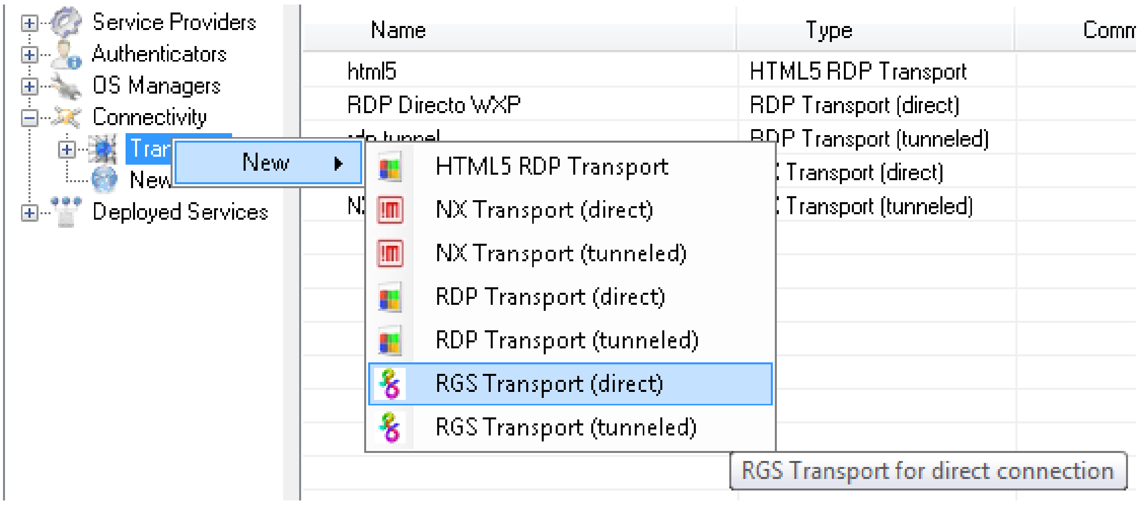
Allow Printers: ☑

Allow Drives: ☑

Allow Serials: ☑

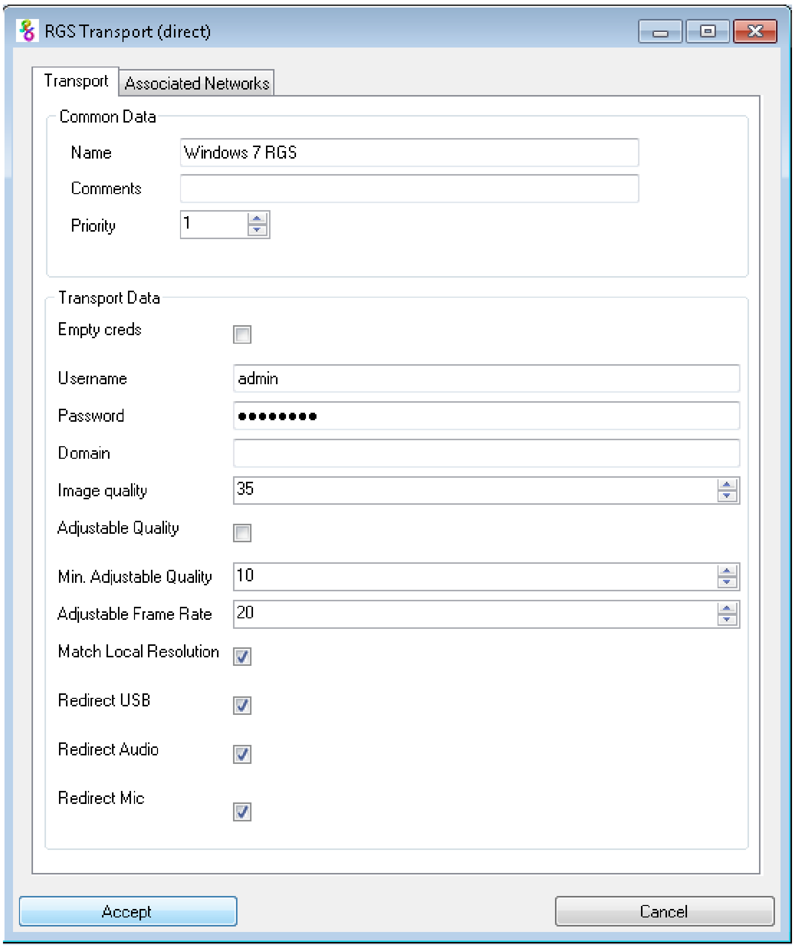
5.6.6 RGS Transport (direct)

A "RGS Transport (direct)" allows access to Windows virtual desktops through RGS protocol (the virtual machines and the connection clients must have RGS service installed).



For the RGS transport (direct), we configure the transport name and we indicate whether we want to redirect specific credentials to the virtual desktop. If the “Empty creds” box is checked, no credential will pass through to the virtual desktop. If we do not check the “Empty creds” box and enter a “username” and “password” (it is also possible to enter a user domain), these credentials will be redirected to the virtual desktop.

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values). We can also indicate the parameters “Image Quality”, “Adjustable Quality”, “Min Adjustable Quality”, “Adjustable Frame Rate”, “Match Local Resolution”, “Redirect USB”, “Redirect Audio” and Redirect Mic”.



RGS Transport (direct)

Common Data

Name: Windows 7 RGS

Comments:

Priority: 1

Transport Data

Empty creds □

Username: admin

Password: ●●●●●●●●

Domain:

Image quality: 35

Adjustable Quality □

Min. Adjustable Quality: 10

Adjustable Frame Rate: 20

Match Local Resolution ☑

Redirect USB ☑

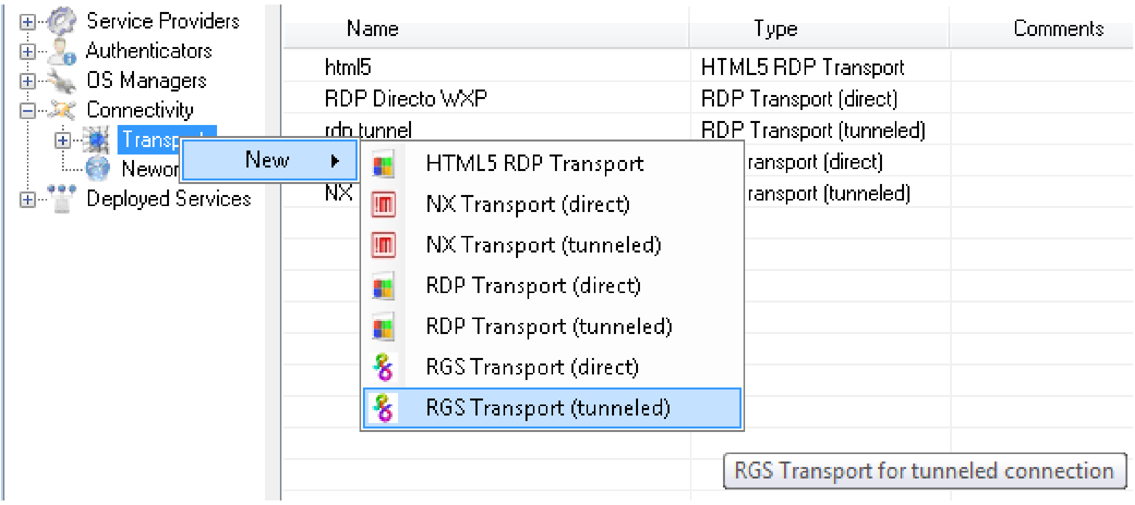
Redirect Audio ☑

Redirect Mic ☑

5.6.7 RGS Transport (tunneled)

A "RGS Transport (tunneled)" allows access to Windows virtual desktops through RGS protocol (the virtual machines and the connection clients must have RGS service installed).

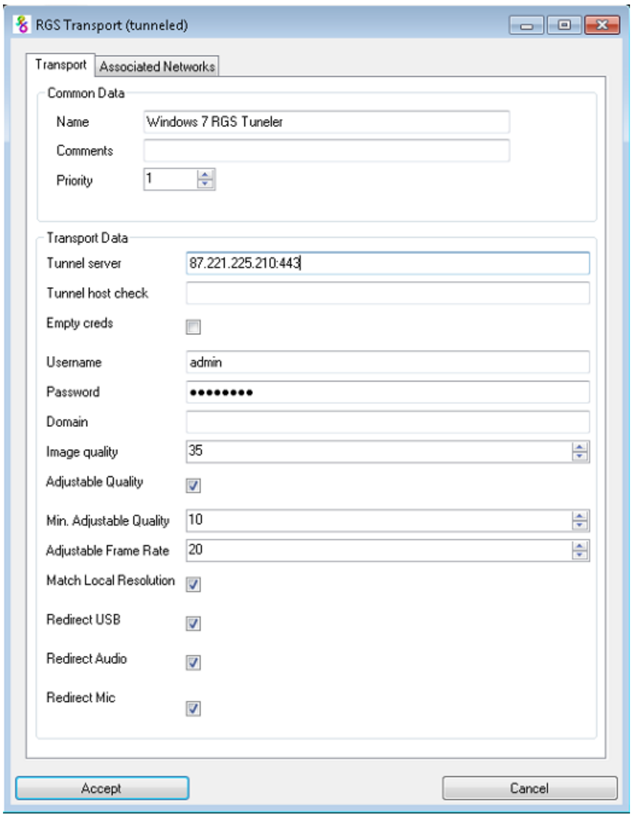
This transport uses UDS Tunneler server to make the connection against the virtual desktops. It must be configured beforehand in order to work properly.



Name	Type	Comments
html5	HTML5 RDP Transport	
RDP Directo WXP	RDP Transport (direct)	
rdp tunnel	RDP Transport (tunneled)	
	ransport (direct)	
NX	ransport (tunneled)	

For the RGS Transport (tunneled), we configure the transport name, the IP address of the UDS Tunneler server and port (“Tunnel server” field) with the format IP_Tunneler:443 (port by default). We indicate whether we want to redirect specific credentials to the virtual desktop, and if the “Empty creds” box is checked, no credential will pass through to the virtual desktop. If we do not check the “Empty creds” box and enter a “username” and “password” (it is also possible to enter a user domain), these credentials will be redirected to the virtual desktop.

We can also indicate the priority of this transport. The lower that priority is, the higher it will appear on the list of transports available in virtual desktop window of each user (this field admits negative values). We can also indicate the parameters “Image Quality”, “Adjustable Quality”, “Min Adjustable Quality”, “Adjustable Frame Rate”, “Match Local Resolution”, “Redirect USB”, “Redirect Audio” and Redirect Mic”.



Common Data

Name: Windows 7 RGS Tuneler

Comments:

Priority: 1

Transport Data

Tunnel server: 87.221.225.210:443

Tunnel host check:

Empty creds: □

Username: admin

Password: ••••••••

Domain:

Image quality: 35

Adjustable Quality: ☑

Min. Adjustable Quality: 10

Adjustable Frame Rate: 20

Match Local Resolution: ☑

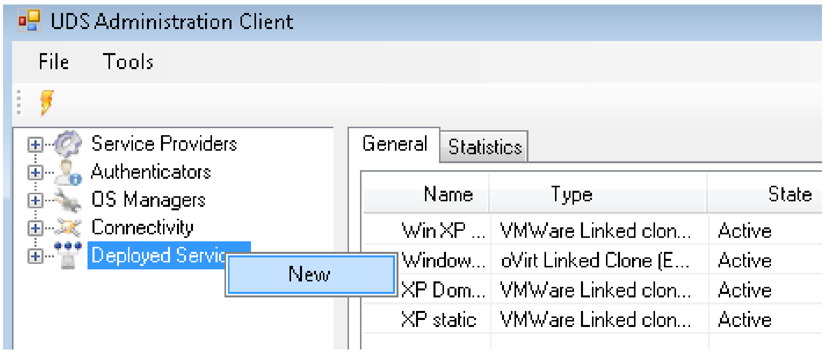
Redirect USB: ☑

Redirect Audio: ☑

Redirect Mic: ☑

5.7 Configuring "Deployed Services"

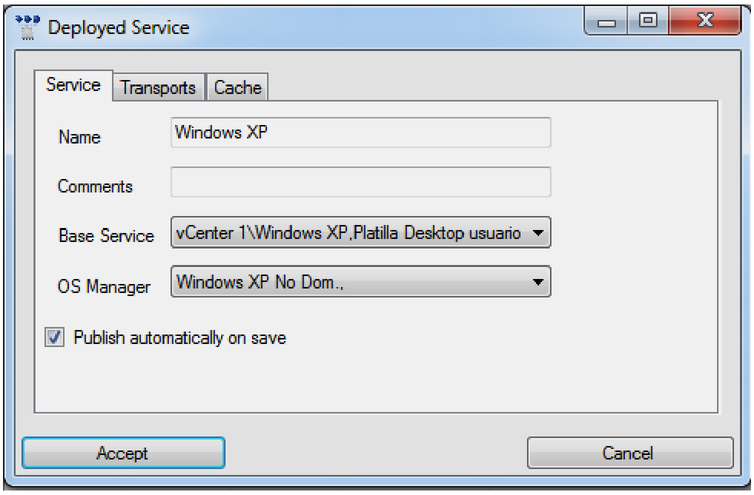
Once the different pieces of UDS platform are configured, it is time to create a "Deployed service". This will be made up by a "Service" created from a "Service Provider", an "OS Manager" and you will also have to indicate one or several transports, one or several access networks (if none is specified, all networks will be allowed) and a user group or groups for accessing this service.



Name	Type	State
Win XP ...	VMWare Linked clon...	Active
Window...	oVirt Linked Clone (E...	Active
XP Dom...	VMWare Linked clon...	Active
XP static	VMWare Linked clon...	Active

To configure a deployed service, enter the service name. It must be a descriptive name since it will be the name that appears to the users for making a connection. Select the base service created in the "Service Providers" section and the appropriate "OS Manager" for the service.

If "Publish automatically on save" is checked, it will inform the system to publish the service when creation thereof is accepted.



Name: Windows XP

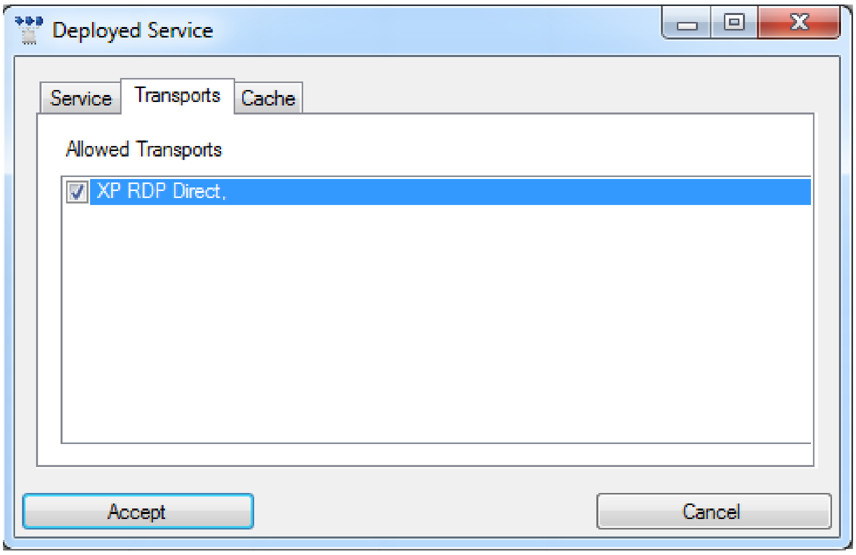
Comments:

Base Service: vCenter 1\Windows XP,Platilla Desktop usuario

OS Manager: Windows XP No Dom.,

☑ Publish automatically on save

In the “Transports” tab, select the appropriate transport for the service. It is possible to select several transports for a single deployed service. The connection order will be determined by the configured priority in each one.

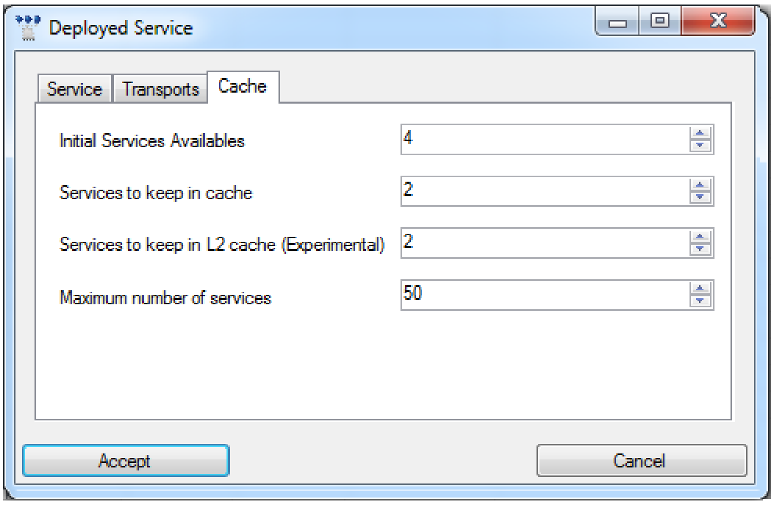


Allowed Transports

☑ XP RDP Direct,

In the “Cache” tab, you can configure how to deploy the service:

- Initial Services Available: Virtual desktops that will be available from the start.
- Services to Keep in cache: Virtual desktops available in the system cache. These desktops will be configured and ready for their assignment.
- Services to Keep in L2 cache: Virtual desktops in sleeping mode. These desktops will be configured and ready for their assignment.
- Maximum number of services: Maximum number of virtual desktops created by the UDS system.



Deployed Service

Service | Transports | Cache

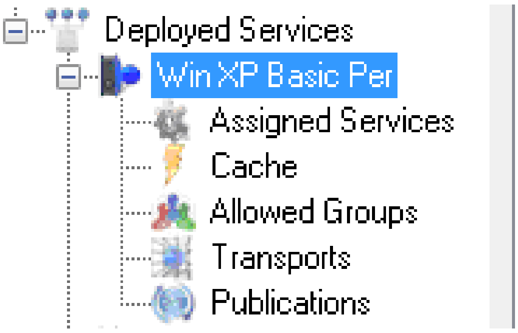
Initial Services Availables: 4

Services to keep in cache: 2

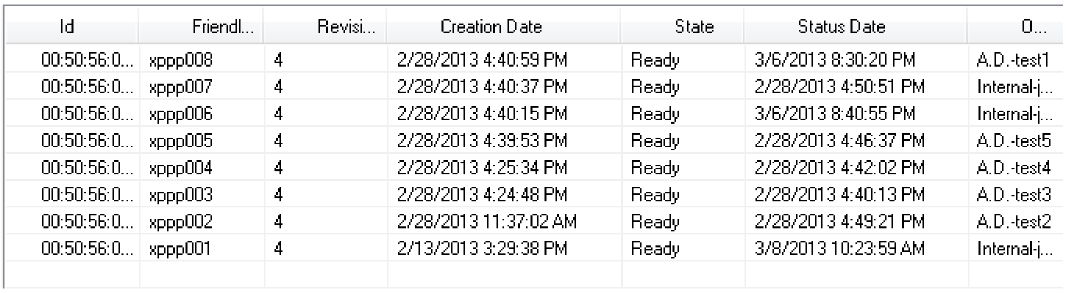
Services to keep in L2 cache (Experimental): 2

Maximum number of services: 50

Once the “Deploy Service” has been created, the following menus will be available:

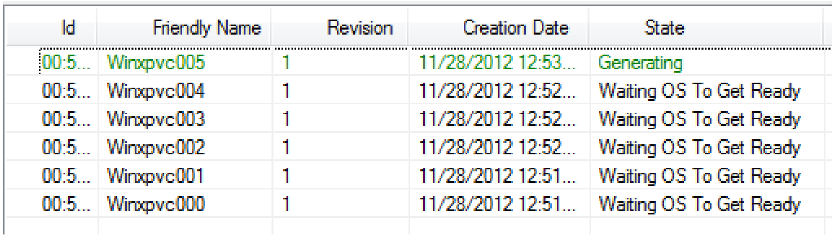


- **Assigned Services :** Virtual desktops assigned to users



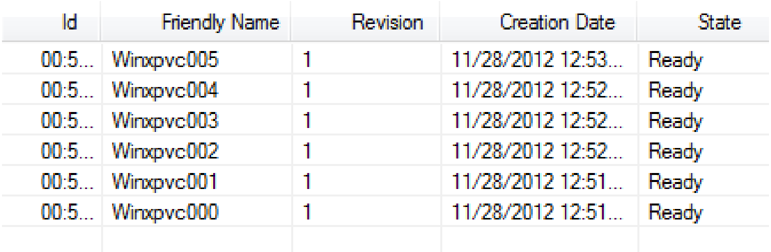
Id	Friendl...	Revisi...	Creation Date	State	Status Date	O...
00:50:56:0...	xppp008	4	2/28/2013 4:40:59 PM	Ready	3/6/2013 8:30:20 PM	A.D.-test1
00:50:56:0...	xppp007	4	2/28/2013 4:40:37 PM	Ready	2/28/2013 4:50:51 PM	Internal-j...
00:50:56:0...	xppp006	4	2/28/2013 4:40:15 PM	Ready	3/6/2013 8:40:55 PM	Internal-j...
00:50:56:0...	xppp005	4	2/28/2013 4:39:53 PM	Ready	2/28/2013 4:46:37 PM	A.D.-test5
00:50:56:0...	xppp004	4	2/28/2013 4:25:34 PM	Ready	2/28/2013 4:42:02 PM	A.D.-test4
00:50:56:0...	xppp003	4	2/28/2013 4:24:48 PM	Ready	2/28/2013 4:40:13 PM	A.D.-test3
00:50:56:0...	xppp002	4	2/28/2013 11:37:02 AM	Ready	2/28/2013 4:49:21 PM	A.D.-test2
00:50:56:0...	xppp001	4	2/13/2013 3:29:38 PM	Ready	3/8/2013 10:23:59 AM	Internal-j...

- **Cache**: Virtual desktops available in the system cache (including level 2 cache machines). These desktops will pass through different states:
    - **Generating**: In this state, the virtual desktops are being created in the virtualization platform.
    - **Waiting OS To Get Ready:** In this state, the desktops are being configured with the parameters indicated in the deployed service.



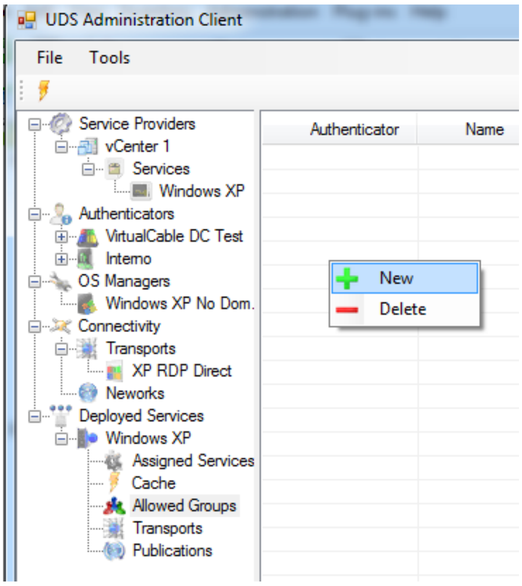
Id	Friendly Name	Revision	Creation Date	State
00:5...	Winxpvc005	1	11/28/2012 12:53...	Generating
00:5...	Winxpvc004	1	11/28/2012 12:52...	Waiting OS To Get Ready
00:5...	Winxpvc003	1	11/28/2012 12:52...	Waiting OS To Get Ready
00:5...	Winxpvc002	1	11/28/2012 12:52...	Waiting OS To Get Ready
00:5...	Winxpvc001	1	11/28/2012 12:51...	Waiting OS To Get Ready
00:5...	Winxpvc000	1	11/28/2012 12:51...	Waiting OS To Get Ready

- **Ready:** When a virtual desktop is found in this state, it means that it is ready for use.

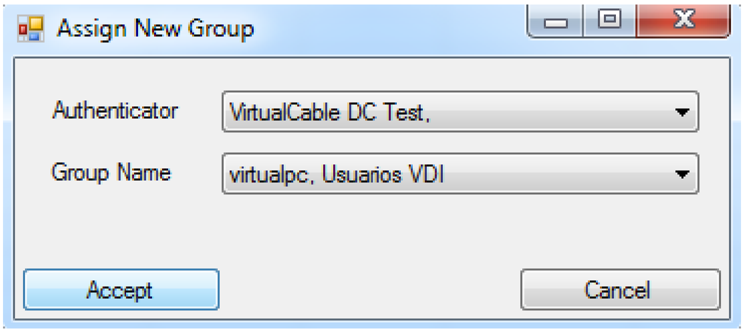


Id	Friendly Name	Revision	Creation Date	State
00:5...	Winxpvc005	1	11/28/2012 12:53...	Ready
00:5...	Winxpvc004	1	11/28/2012 12:52...	Ready
00:5...	Winxpvc003	1	11/28/2012 12:52...	Ready
00:5...	Winxpvc002	1	11/28/2012 12:52...	Ready
00:5...	Winxpvc001	1	11/28/2012 12:51...	Ready
00:5...	Winxpvc000	1	11/28/2012 12:51...	Ready

- **Allowed Groups:** In order to allow users to connect, you must assign access groups or metagroups. These groups or metagroups must be created in the Authenticators section. We can assign one or several access groups or metagroups to each deployed service.



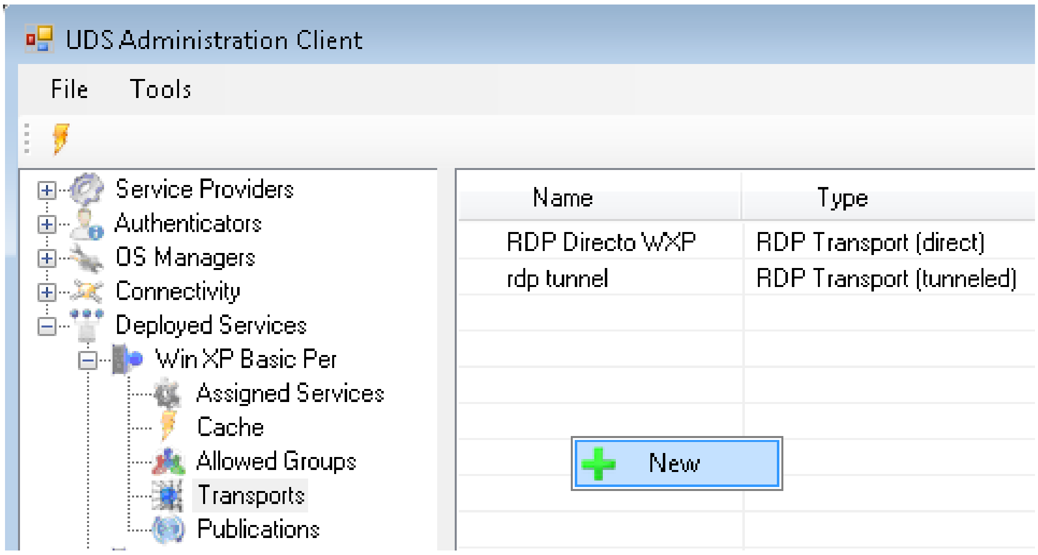
We select the Authenticator and based on the choice we select the Group Name.



Authenticator: VirtualCable DC Test,

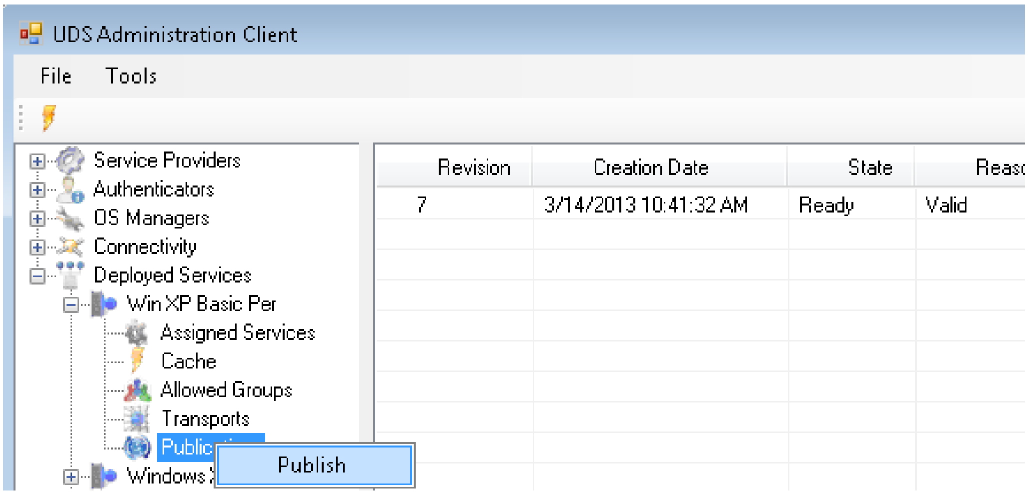
Group Name: virtualpc, Usuarios VDI

- **Transports:** The transports configured for the deployed service are shown. It is also possible to add new transports (previously added in the "Transports" section) from this menu.



Name	Type
RDP Directo WXP	RDP Transport (direct)
rdp tunnel	RDP Transport (tunneled)

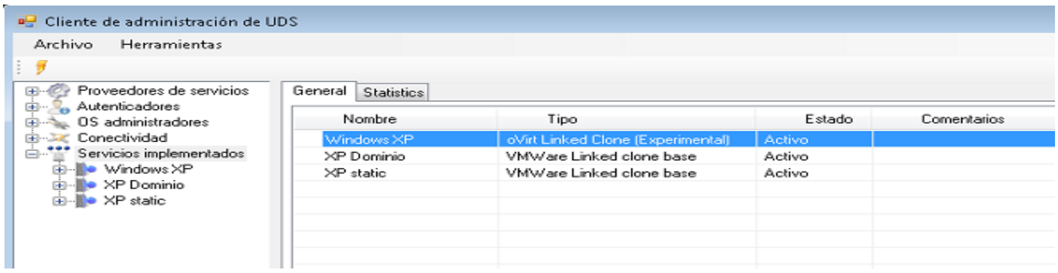
- **Publications**: From this menu, we will be able to publish a new service. Once this publication process has been completed, the entire system cache will be recreated with the new Linked Clones based on this last publication.



Revision	Creation Date	State	Reas
7	3/14/2013 10:41:32 AM	Ready	Valid

5.7.1 “General” Tab

This tab shows a summary of the situation of all of the implemented services that are managed by UDS, indicating the status and type of service provider.

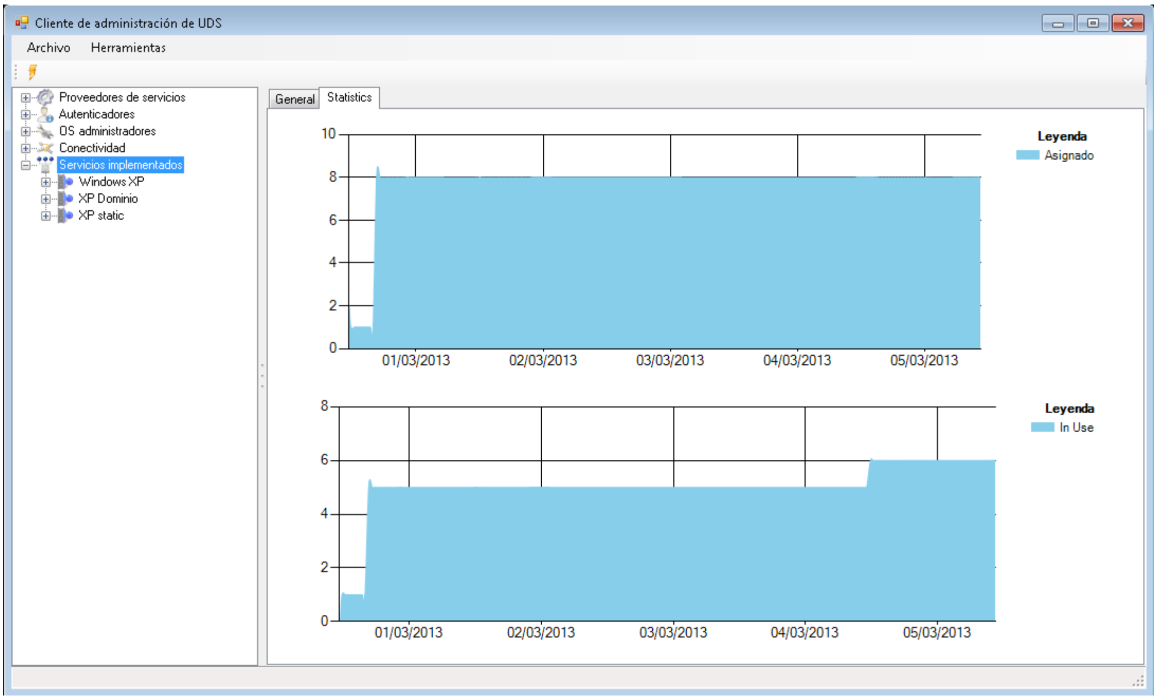


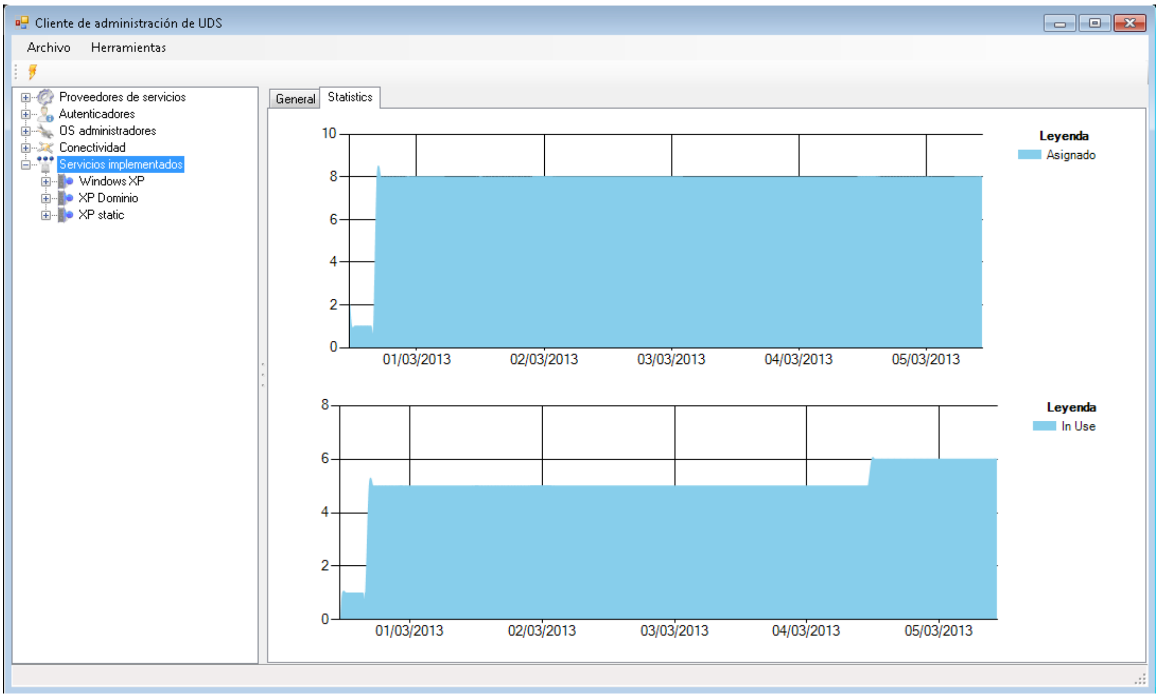
Nombre	Tipo	Estado	Comentarios
Windows XP	oVirt Linked Clone (Experimental)	Activo	
XP Dominio	VMWare Linked clone base	Activo	
XP static	VMWare Linked clone base	Activo	

5.7.2 "Statistics" Tab

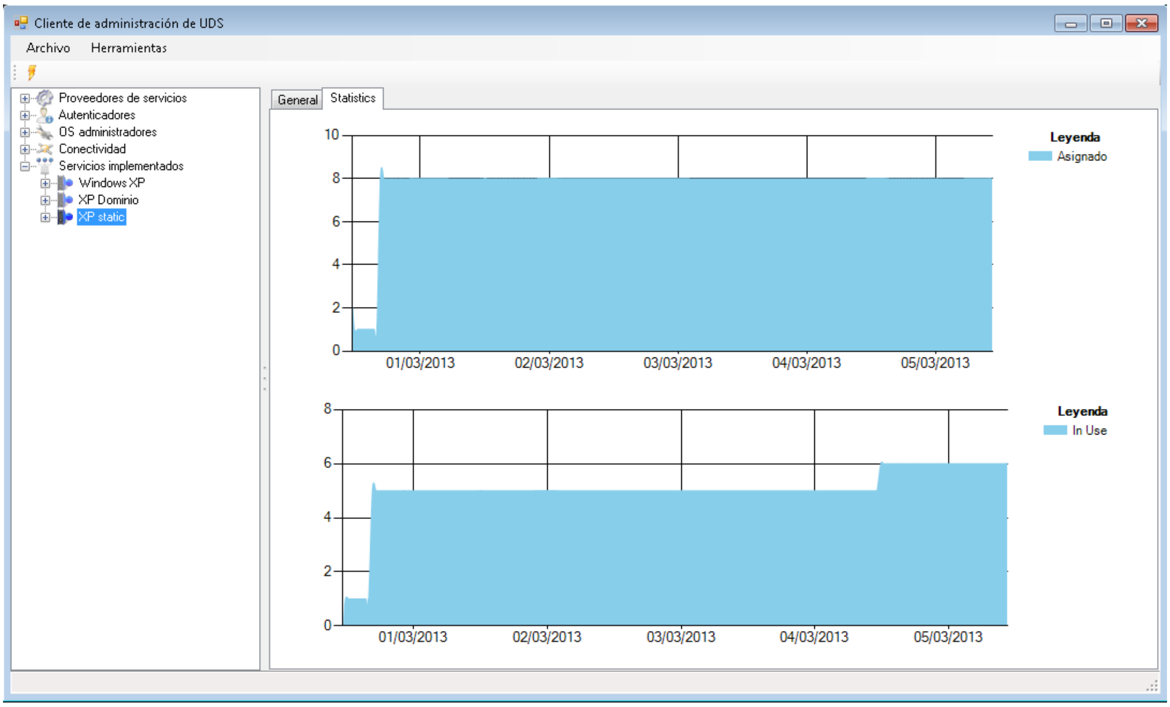
UDS extracts information from the active services implemented and generates statistics of the assigned PCs and PCs currently in use.

These statistics are generated in a general way from all of the implemented services by clicking on "Deployed Services." The statistic shown, in this case, is an average formed by all of the active implemented services.



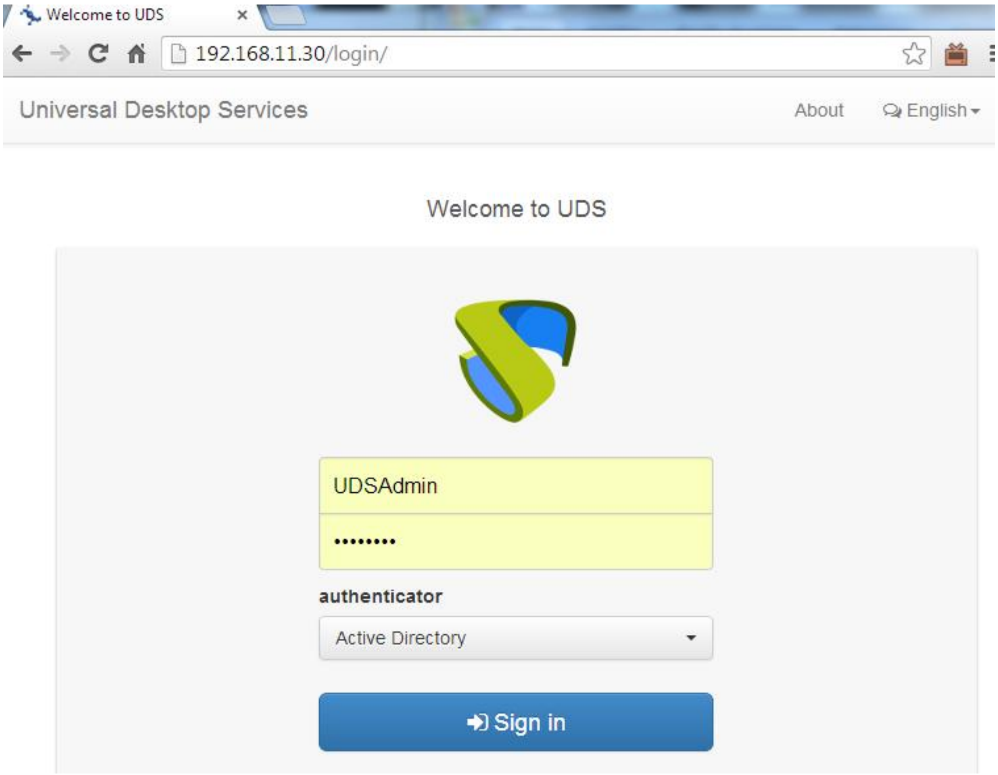

Cliente de administración de UDS
Archivo
Herramientas
Proveedores de servicios
Autenticadores
OS administradores
Conectividad
Servicios implementados
Windows XP
XP Dominio
XP static
General
Statistics
10
8
6
4
2
0
01/03/2013
02/03/2013
03/03/2013
04/03/2013
05/03/2013
Leyenda
Asignado
8
6
4
2
0
01/03/2013
02/03/2013
03/03/2013
04/03/2013
05/03/2013
Leyenda
In Use

It is also possible to view the statistics of a specific implemented service by clicking on the name of the implemented service of which you would like to view the information.



6 DEPLOYING VIRTUAL DESKTOPS WITH UDS

Once one or several Service Pools are available, we can initiate a connection to a virtual desktop. We will gain access via a web browser (the supported browsers are Internet Explorer, Chrome and Firefox) to the broker address or name, enter a valid username and password and select the authenticator.



UDSAdmin

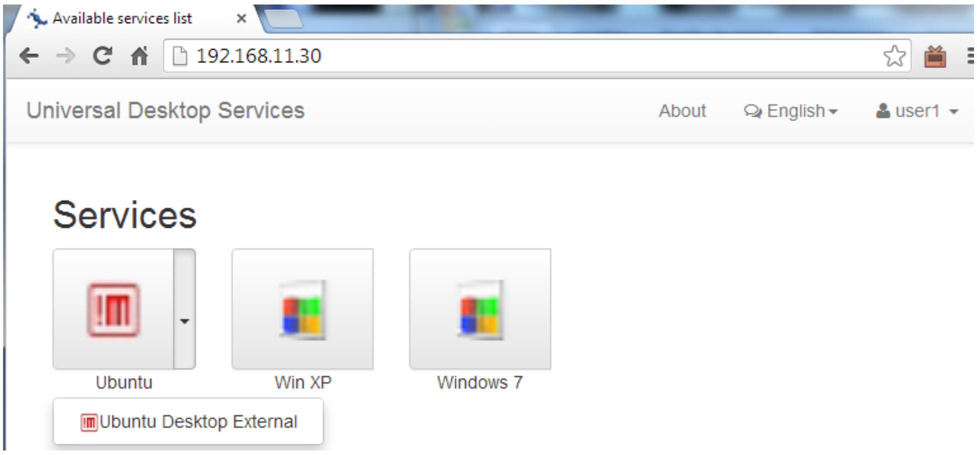
••••••••

authenticator

Active Directory

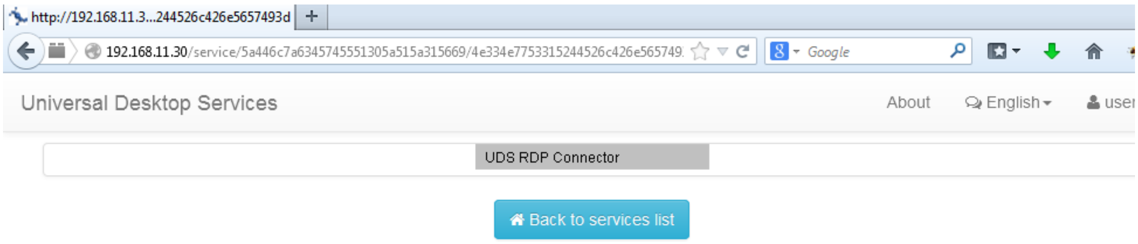
In the available services screen, the services that the user with which we initiated the session in the UDS system has access to will appear. Click on the one with which we want to start the connection.

By default, if we click directly on the service image we will make the connection with the Transport which has the lower priority. If we have configured several Transports, a pull-down menu will appear, where we will be able to select the Transport we are going to use to connect to the virtual desktop.

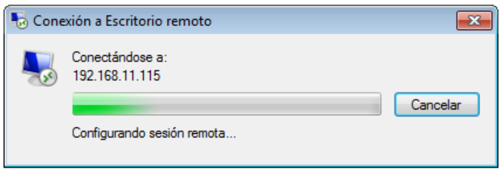


To start the connection to the virtual machine, you must have Java installed in the client machine to start the connection for all the transports except HTML5. For connections by NX we will have to have the software NX installed and for connections by RGS you must have the RGS software installed.

RDP connection example:



http://192.168.11.30/service/5a446c7a6345745551305a515a315669/4e334e7753315244526c426e565749

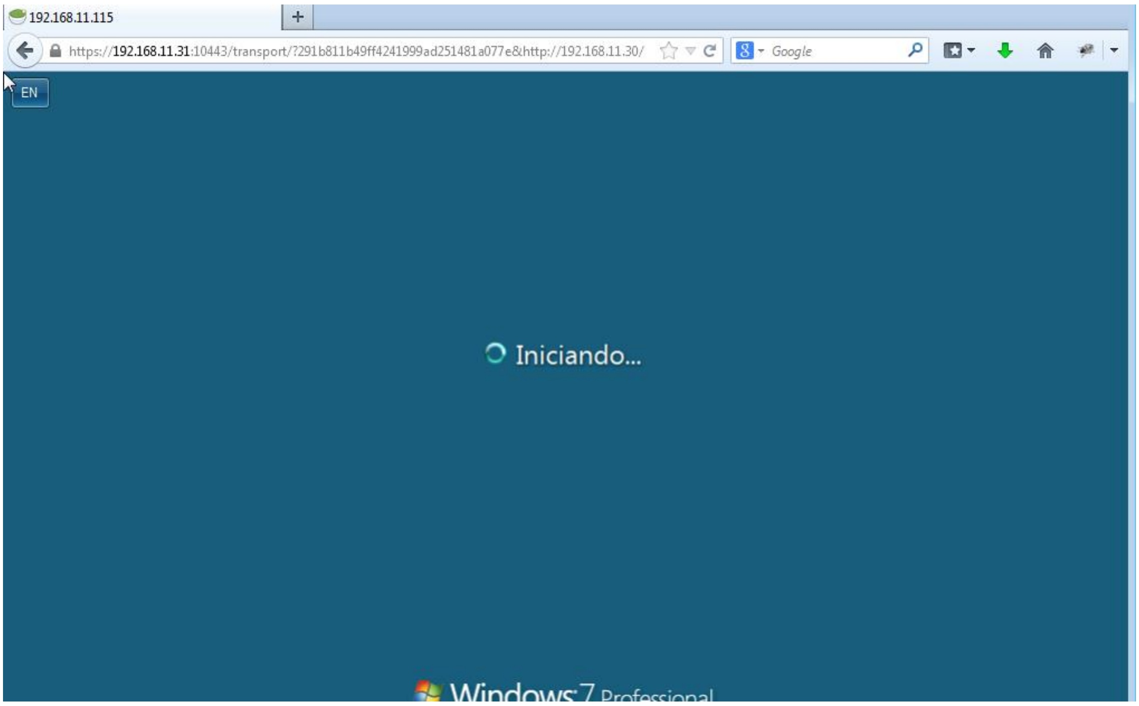


Conexión a Escritorio remoto

Conectándose a:
192.168.11.115

Configurando sesión remota...

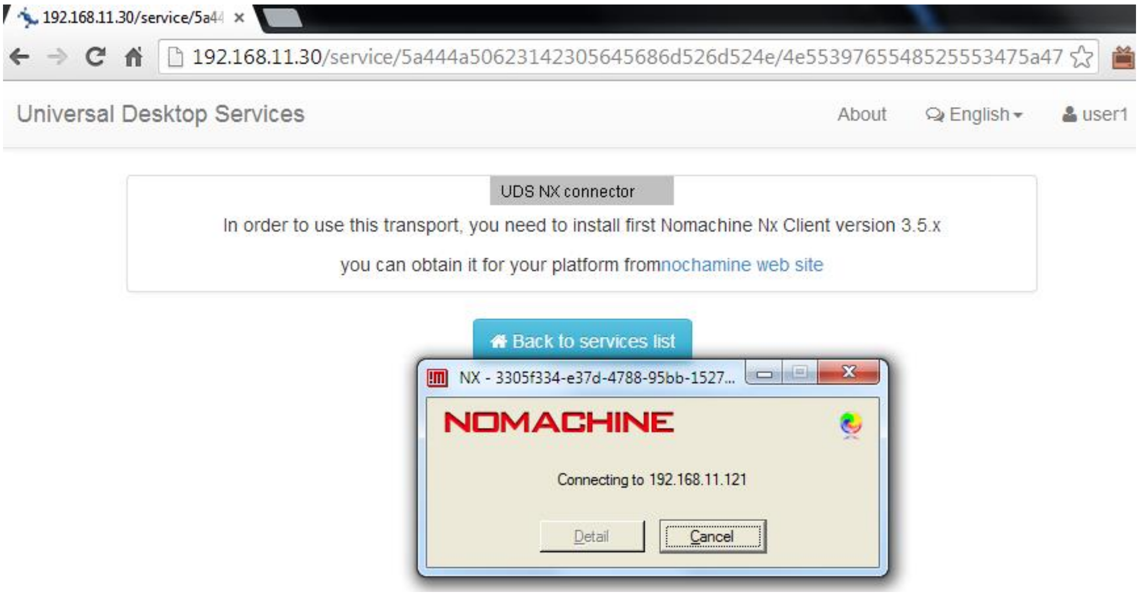
HTML5 connection example:



https://192.168.11.31:10443/transport/?291b811b49ff4241999ad251481a077e&http://192.168.11.30/

Iniciando...

NX connection example:



192.168.11.30/service/5a444a506231423056456864d526d524e/4e5539765548525553475a47

Universal Desktop Services

UDS NX connector

In order to use this transport, you need to install first Nomachine Nx Client version 3.5.x

you can obtain it for your platform fromnochamine web site

NX - 3305f334-e37d-4788-95bb-1527...

Connecting to 192.168.11.121

Once the connection is made, the virtual desktop will be available for use.

7 UDS ADVANCED CONFIGURATION

UDS provides a series of advanced parameters which will define the running of the system. These parameters will be responsible for defining aspects like security, connectivity, operating mode… both of UDS system and its communication with the virtual platforms (VMware, oVirt, RHEV, Hyper-V) registered on UDS.

In this document we will only show the system variables that are considered to be the more useful ones to manage the virtual desktops. When using the variables which aren't mentioned here, it is recommended not to modify the default values, as some of them indicate how the system has to work (number of simultaneous tasks, time for tasks execution, programmed checks, etc…) and a wrong parameter modification may completely stop the system or make it work in a wrong way.

Once the values of one of the UDS advanced configuration variables have been modified, it will be necessary to reboot the UDS Server so that the changes will be applied.

If you want to modify any value which isn't included in this section, we recommend you to contact the UDS Enterprise support team in order to verify that change and confirm that it doesn't affect in a negative way to the running of UDS system.

7.1 Access to UDS advanced parameters

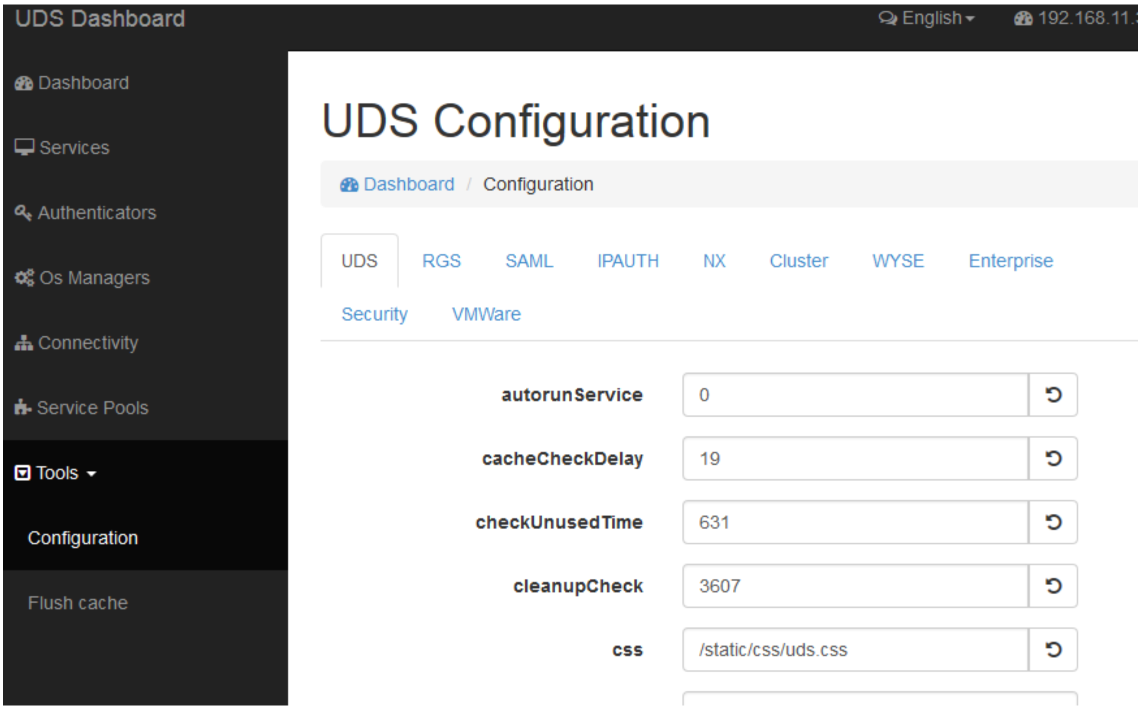
In UDS Enterprise 1.5 version you can access the advanced parameters in two different ways:

- Web access.
- Access through administration client.

"UDS Server" must be rebooted whenever you modify any variable so that the system will make these changes, both if the changes have been made using web administration or administration client.

7.1.1 Web administration

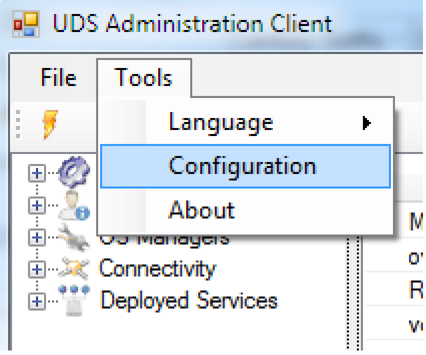
In order to access the UDS advanced configuration parameters through web administration, access "Tools" section and click "Configuration":

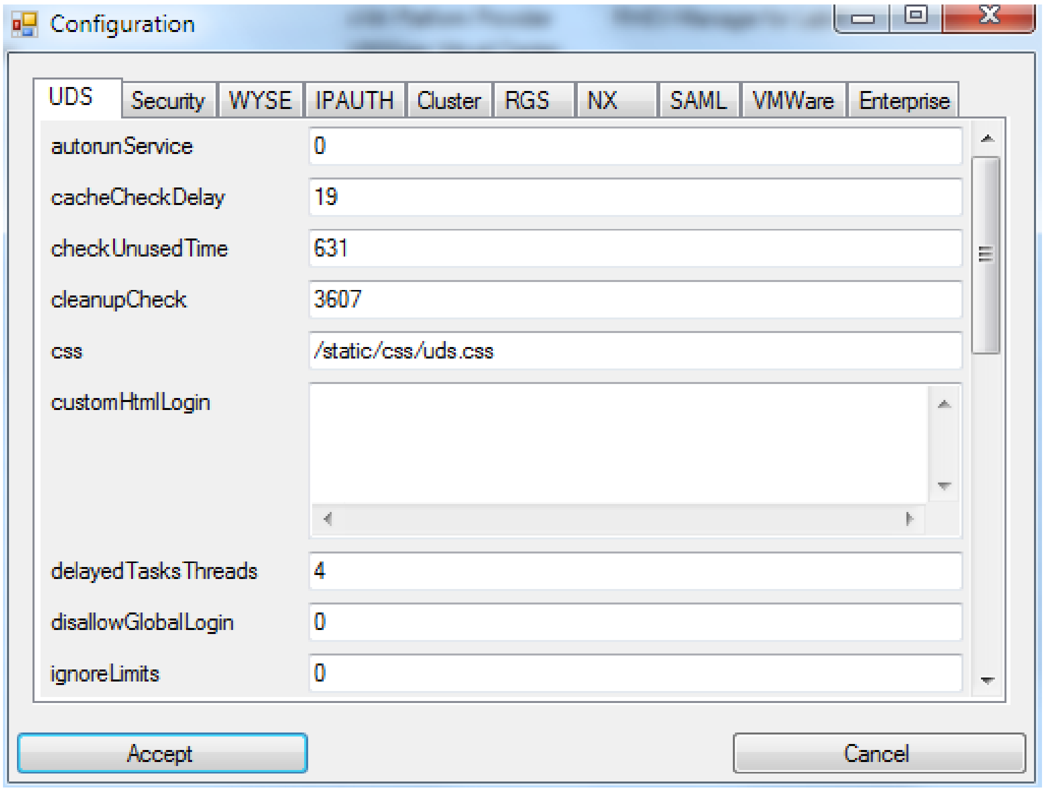


autorunService	0
cacheCheckDelay	19
checkUnusedTime	631
cleanupCheck	3607
css	/static/css/uds.css

7.1.2 UDS administration client

In order to access the UDS advanced configuration parameters through the administration client, access “Tools” section and click “Configuration”:





autorunService	0
cacheCheckDelay	19
checkUnusedTime	631
cleanupCheck	3607
css	/static/css/uds.css
customHtmlLogin	
delayedTasksThreads	4
disallowGlobalLogin	0
ignoreLimits	0

7.2 UDS advanced parameters

7.2.1 UDS

The most important parameters regarding UDS internal procedures, appearance and communication with the hypervisor platforms.

AutorunService = It makes the direct access to the virtual desktop when a user only has one service assigned (0 = off 1 = on).

If you turn this parameter on, the users who only have one virtual desktop assigned will connect to it straightaway, the window where you select the service won't appear and the first configured transport will be used.

Default value 0.

CustomHtmlLogin= HTML code for partial customization of the UDS login page.

The code you enter will appear under the user login box in the UDS access dashboard.

Empty by default.

DisallowGlobalLogin = If it is turned on, the entire authenticators list won't appear (0 = off 1 = on).

If this variable is turned on, you must use a "short name" to see an authenticator and allow user access to the system.

Default value 0

RedirectToHttps = It automatically redirects UDS Enterprise access from http to https (0 = off 1 = on).

Default value 0

SessionExpireTime = It indicates the maximum time a user session will be opened after publishing something new. After, the user session will be closed and the system will delete the service.

Default value 24 hours.

StatsDuration = It indicates how long the system will keep the statistics.

Default value 365 days.

UDSTheme = Name of the folder which contains the HTML templates for the login Windows, user, preferences, downloads, etc…

Default name html5.

7.2.2 RGS

Find below the description of the parameters related to the RGS Transport:

DownloadUrl = Web address to download RGS software.

TunnelOpenedTime = Maximum time the tunnel will wait for the RGS connection to be opened.

If the connection isn't carried out in the time indicated in this variable, it will be canceled and you will have to make the connection again (if you make the connection using slow clients, it is recommended to increase this value)

Default value 30 seconds.

7.2.3 SAML

Find below the description of the parameters related to the SAML authenticator:

Global logout on exit = It indicates the logout mode (0 = off 1 = on).

If it is enabled, when you logout from UDS you also logout from SAML.

Default value 0

IDP Metadata Cache = Time the IDP.m searched metadata are kept.

Default value 86400 seconds (24 hours).

Organization Display Name = Organization name displayed.

Organization Name = Organization name.

Organization URL = Organization web address.

User cleanup = It indicates how often the system cleans up the users without activity.

If a user remains inactive for the time indicated in this variable, the system will delete it.

Default value 2592000 seconds (30 days).

7.2.4 IPAUTH

These variables are inactive in this UDS Enterprise version.

7.2.5 NX

Find below the description of the parameters related to the NX Transport:

DownloadUrl = Web address to download NX software.

DownloadUrlMACOS = Web address to download NX software for MAC.

7.2.6 CLUSTER

These variables are inactive in this UDS Enterprise version.

7.2.7 WYSE

Find below the description of the parameters related to the connection with Wyse clients:

Autoconnect = It allows the automatic connection of the device.

Default value no.

Colors = It defines the quality of the colours displayed during the connection.

Default value High.

DefaultUser = Default user redirected to the device.

Default value UDS.

Language = Device language.

Default value us.

Privilege = User privilege level.

Default value NONE.

For more details about these parameters see Wyse official documentation or the following reference guide: http://www.freewysemonkeys.com/downloads/wtos/Wyse%20Thin%20OS%2064%20Parameters.pdf

7.2.8 ENTERPRISE

Find below the description of the parameters related to UDS Enterprise subscription:

Serial Number = Subscription activation code.

During the UDS Server configuration you must indicate a valid serial number. Using this variable you can update or change it.

7.2.9 SECURITY

Find below the description of the parameters related to UDS system security:

AdminIdleTime = It indicates how long an administrator session will be opened. After this period, you must authenticate yourself again in the system.

Default value 14400 seconds (4 hours).

AllowRootWebAccess = It allows the root user login via web (0 = off 1 = on).

The modification of this variable doesn't affect the root user access through the administration client.

Default value 1.

RootPass = Root user password, previously indicated in the UDS Server configuration script.

SuperUser = User with UDS system administration rights.

By default: root

Trusted Hosts = Hosts considered to be secure by UDS. These host can make sensitive requests to UDS, for example tunnelers (it is recommended to modify this variable so that the only displayed option is the list of tunnelers).

By default * (all allowed), it admits addresses range values.

UserSessionLength = It indicates how long the user session will be opened. After this period, it will be necessary for the user to authenticate himself again in the system.

Default value 14400 seconds (4 hours)

7.2.10 VMWARE

Find below the description of the parameters related to VMware vSphere virtual platform:

MaxRetriesOnError = Number of times and operation is retried in case VMware reports an error to UDS system.

Default value 63 retries.

MinUsableDatastoreGB = Minimum free space in a datastore to create the virtual desktops.

If the VMware platform datastores selected to create services in UDS have less free space than the value of this variable, the virtual desktops won't be created. Once this value is modified or the needed space is available, the system will work properly.

Default value 30.

8 ABOUT VIRTUALCABLE

VirtualCable sells UDS via a subscription model, including product support and segment updates by number of users.

VirtualCable also offers a broad portfolio of professional services in order to help its clients in both installation processes and UDS configuration and in virtualization projects with other platforms.

For more information, visit www.udsenterprise.com or contact us via email: sales@udsenterprise.com.

All the proper names of programs, operating systems, hardware, etc. that appear in this document are registered trademarks of their respective companies or organizations.

-END OF DOCUMENT-