Linux application Virtualization with UDS Enterprise

UDS
ENTERPRISE

3.6

Linux Application Virtualization with UDS Enterprise

Index

Introduction

The UDS Enterprise VDI and vApp connection broker allows the deployment and management of virtualized Linux application sessions through the X2Go connection protocol. These applications can be assigned to user groups.

Access to Linux virtual application sessions can be done through Windows and Linux OS connection clients.

Users will run the applications in independent Linux virtual machines autogenerated by UDS Enterprise, based on a template machine (where the applications must be installed). If the same user accesses several applications, all those applications will run on the same virtual desktop, thus achieving significant resource optimization.

For the correct operation and integration of UDS Enterprise with X2Go, it is necessary to perform a series of tasks that are detailed in this document.

Necessary elements

To configure the different elements that will make up the base machine (template) to be used with UDS Enterprise to serve virtual applications, you need:

1. Virtualization platform

It is necessary to have a virtualization platform that integrates with UDS Enterprise so that the connection broker can deploy self-generated virtual machines. These virtual machines will be the ones that will run the applications for the different users.

The virtualization platforms supported by UDS Enterprise to deploy Linux applications are: Citrix XenSenver / XCP-ng, Microsoft Azure, Microsoft HyperV, oVirt / RHEV, VMware vSphere, VMware vCloud, Nodeweaver, OpenNebula, OpenStack, etc ...

The virtual platform must have sufficient resources (vCPU, vRAM and disk) to be able to run the Linux virtual desktops that will provide the virtual applications.

2. Linux OS Image

This VM will be the one that UDS will use as the base machine (template) to deploy the virtual desktops on which the users will run their virtual applications.

It is recommended to use an Ubuntu distribution as a template. In this example, a minimal image of Ubuntu 22.04 will be used, available from the official Ubuntu repository: https://ubuntu.com/download/desktop

Ubuntu 22.04.1 LTS (Jammy Jellyfish)

Select an image

Ubuntu is distributed on two types of images described below.

Desktop image

The desktop image allows you to try Ubuntu without changing your computer at all, and at your option to install it permanently later. This type of image is what most people will want to use. You will need at least 1024MiB of RAM to install from this image.

64-bit PC (AMD64) desktop image

Choose this if you have a computer based on the AMD64 or EM64T architecture (e.g., Athlon64, Opteron, EM64T Xeon, Core 2). Choose this if you are at all unsure.

3. X2Go protocol

The connection protocol that will allow access to Linux applications will be X2Go.

The UDS Enterprise team has created a script that automates and simplifies the entire X2Go installation process. This script (we have generated one for the server part, which we installed in the template, and another for the client part of Linux computers, which is installed in the connection client) can be downloaded from this repository:
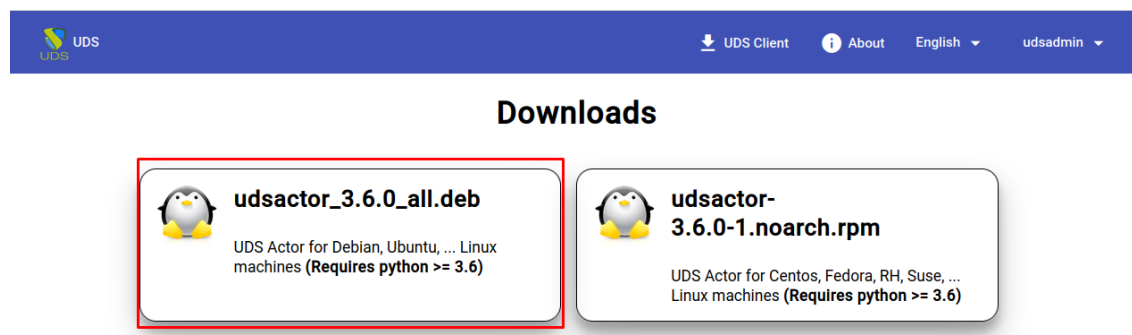
http://images.udsenterprise.com/files/X2GO/

Index of /files/X2GO

Name	Last modified	Size	Description
Parent Directory		-	
X2GO-client.sh	2017-09-25 17:22	627	
X2GO-server.sh	2017-09-25 17:22	689	
X2GO.tar.gz	2017-09-29 09:44	17K	

Apache/2.4.25 (Debian) Server at images.udsenterprise.com Port 443

4. UDS Actor

It is necessary to have the latest stable version of the UDS Actor to take care of the reconfiguration of all Linux virtual desktops generated automatically by the UDS Server. To download the UDS Actor, it is necessary to validate in the UDS login window with a user with administrator permissions. Display the user menu and access downloads. In this window the Actor will be downloaded for Linux machines and specifically for distributions based on Debian (.deb).


5. Various

It is necessary to have Internet access on the template machine to be able to install both the Ubuntu OS and the X2Go connection protocol. It is also necessary to have a DHCP server on the network where the virtual desktops are generated.

Configuring the Linux template and connection client

All the necessary tasks to be carried out both on the Linux base machine (template) and on the connection client (equipment that will access the Linux virtual applications) are detailed below.

1. Linux OS

We will start by creating the VM to be used as the base machine (template). Once the Ubuntu 20.04 image has been downloaded from the official repository indicated in the previous section, you can proceed with the installation:
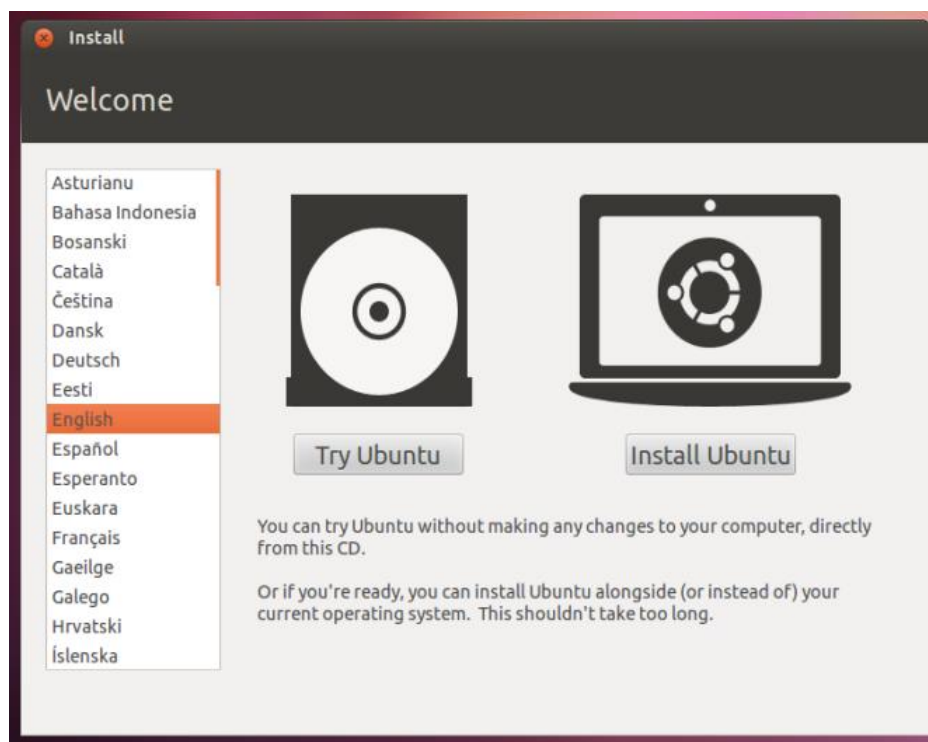

The installation will be carried out with the default values and indicating some resources (disk, vRAM and vCPUs) according to the requirements of the applications that will be installed on the machine.

Once the OS installation is complete, it is important to install the virtual machine drivers corresponding to the virtualization platform. In this example the "open tools" will be installed, since it will run on a VMware vSphere virtual platform.

```
sudo apt-get install open-vm-tools
```

It is also recommended to disable automatic updates and, if you need to update the OS, do it manually.
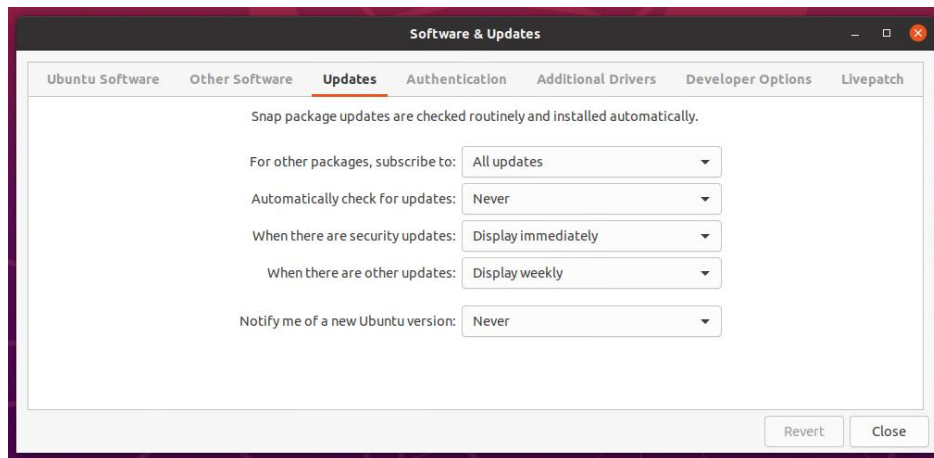
2. X2Go Protocol - Server

Before proceeding with the installation of X2Go, we will need to update our repositories:

sudo apt-get update

Once updated, we will proceed to the installation of X2Go (Server part) through the script provided by VirtualCable (it is necessary to execute the script with a graphical environment session running). The script will be copied to our Xubuntu template:

wgethttp://images.udsenterprise.com/files/X2GO/X2GO-server.sh

We locate where the file was downloaded.

The file is executed:

Sudo bash /home/uds/X2GO-server.sh

When all the components are installed, you can proceed with the installation of X2Go on the connection client computer.

3. UDS Actor on Linux

The UDS Actor is downloaded for Debian-based distributions.
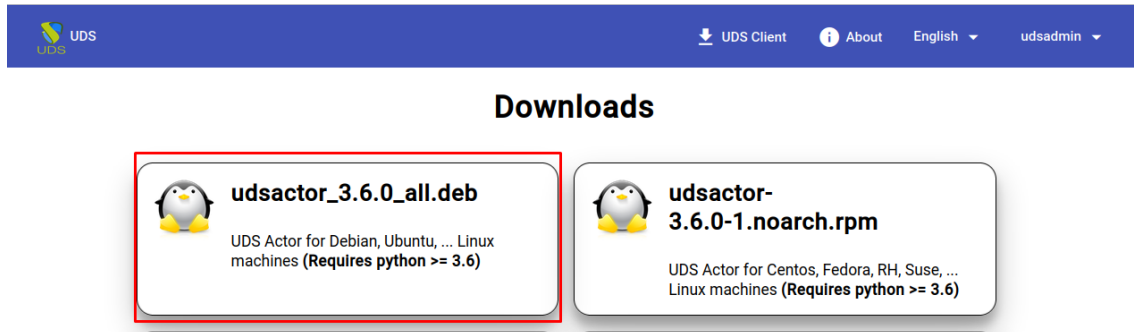
Python version 3.6 will be required for the UDS Actor installation.
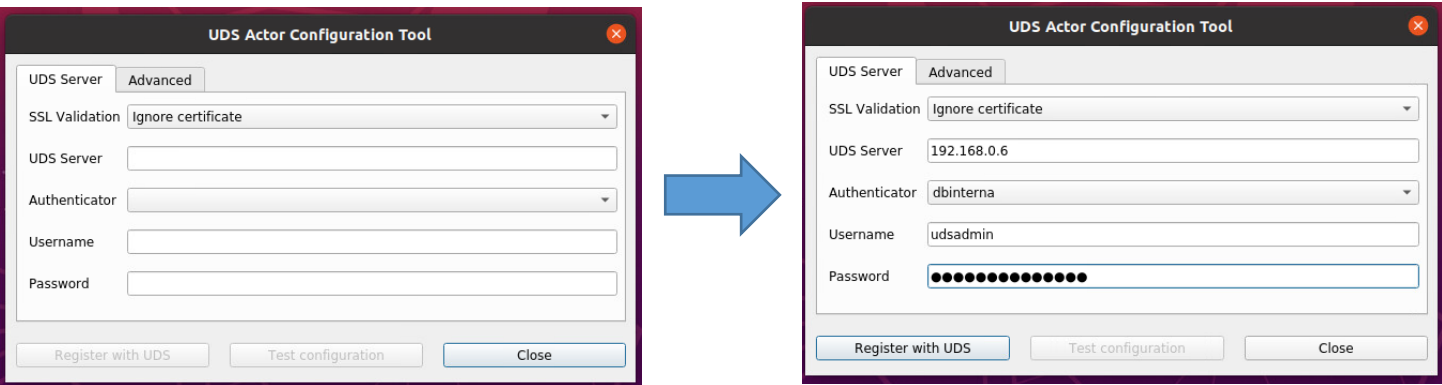
The .deb package that contains the UDS Actor is installed:

In the following window, the information of the UDS Enterprise platform will be requested:

In case any parameter needs to be modified, it can be done from this window or also by editing

the file: /etc/udsactor/udsactor.cfg

4. X2Go Protocol - Connection Client

LINUX

Now the client from which the connection will be made will be prepared. This part will not be done in the template. In the case of having a Linux connection client, we will have to update the system:

```
sudo apt-get update
```


Once updated, you can proceed to the installation of X2Go on the client through the script provided by VirtualCable (it is necessary to run the script with a graphical environment session running). We will have to copy the script in our Linux client:

```
wget http://images.udsenterprise.com/files/X2GO/X2GO-client.sh
```


Locate the script and proceed with its execution:

WINDOWS

If you have a Windows client, you will have to download the X2GO client from the following link:
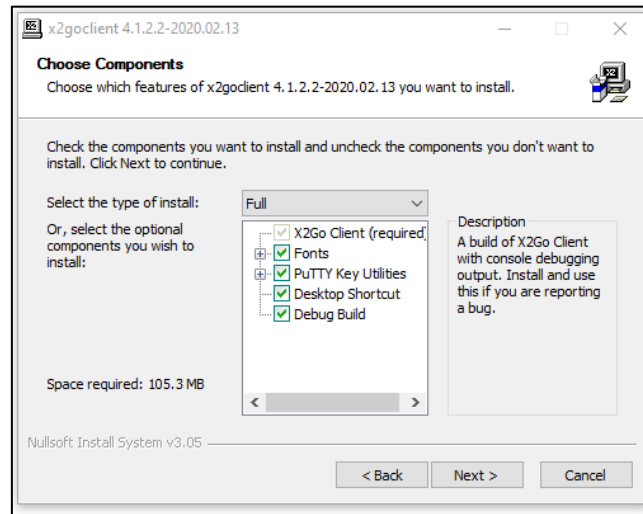https://wiki.X2Go.org/doku.php
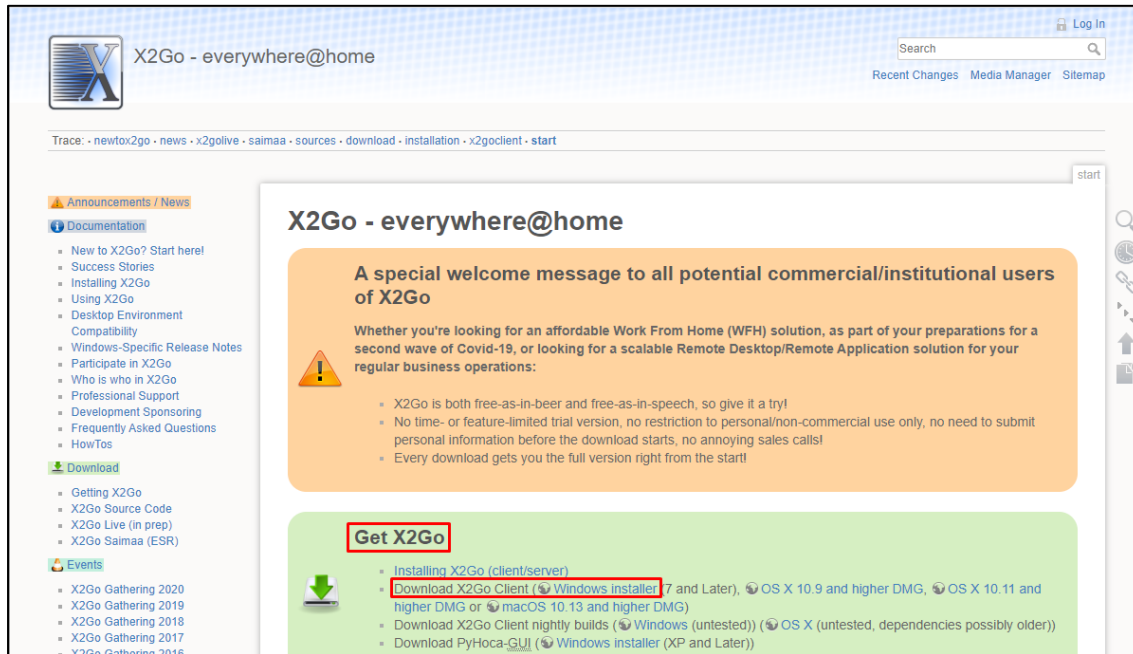
After completing the download, the X2Go client installer runs, following the appropriate steps for each situation:
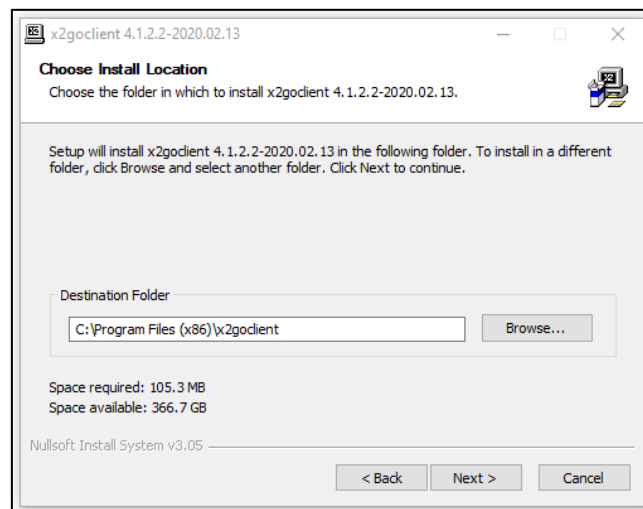
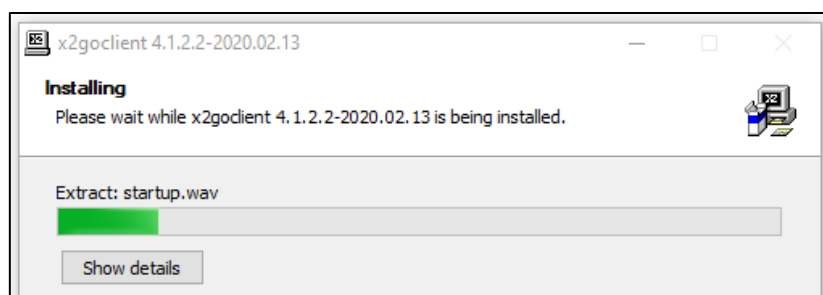You will have to accept the terms of the license of use by clicking on agree:
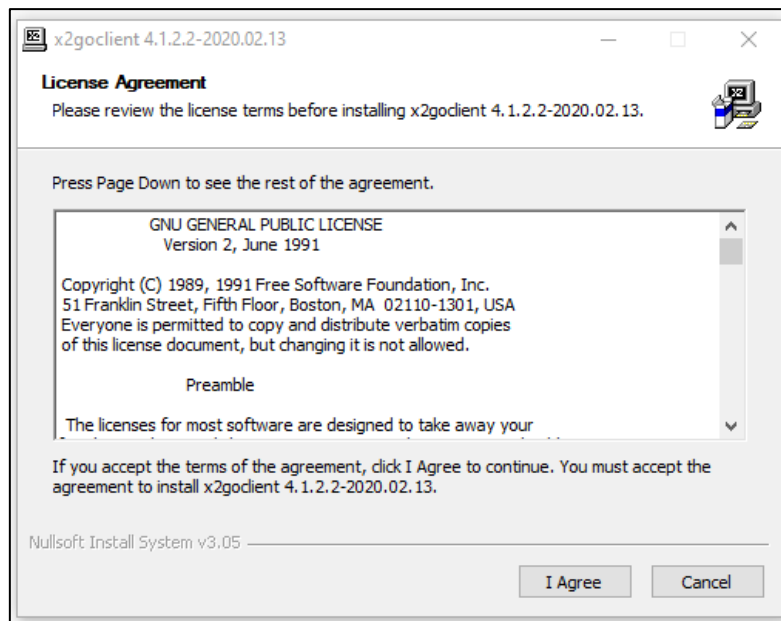
The components to install are selected. In this case, the type of installation will be selected as "Full" and click on "Next":
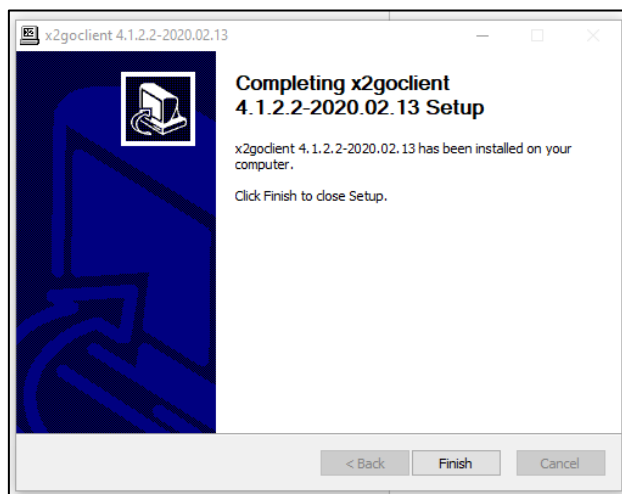

The path in which the installation is to be carried out will be selected and we select "Next":


Select the folder from the start menu where you want to install:

After finishing the installation, it will indicate that it was successful:

Once the X2Go connection protocol is installed in our connection client, it is advisable to test that the connection can be made correctly. For this, a remote connection will be made from our client to the previously configured template.
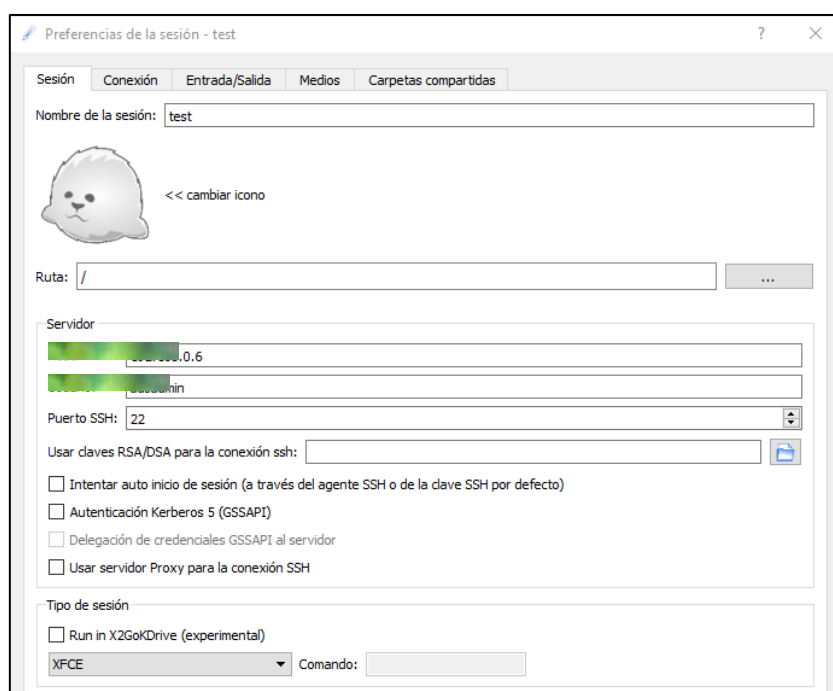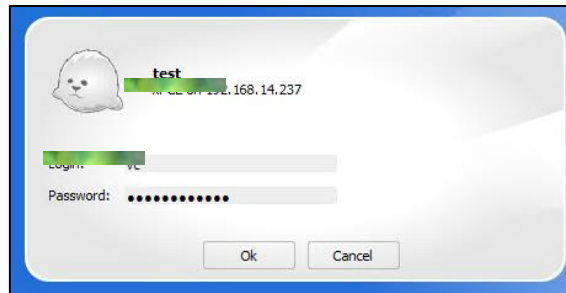

In the "Host" field, enter the IP of the device to which you want to connect, in this case the template.

Enter the username in the "Login" field. You need to select XFCE as "Session type".

Proceed with the login page and enter the password.

If the connection is successful and the remote desktop is displayed, X2Go is working properly so that applications can be virtualized.

NOTE: In client computers, in addition to installing the X2Go client, it will also be necessary to have the UDS client (UDS Plugin) installed. For more details on its installation, check the guide: UDS Enterprise Installation, Administration and User Manual available in the section of documentation from the UDS Enterprise website.

Service publication

Once the Linux template has been prepared (the OS installed, applications to virtualize, X2Go protocol and UDS Actor), we will proceed to configure in UDS and create a new pool of services based on this template.

In order to configure a pool of services and deploy Linux applications, the UDS control panel will be accessed with an administrator user.
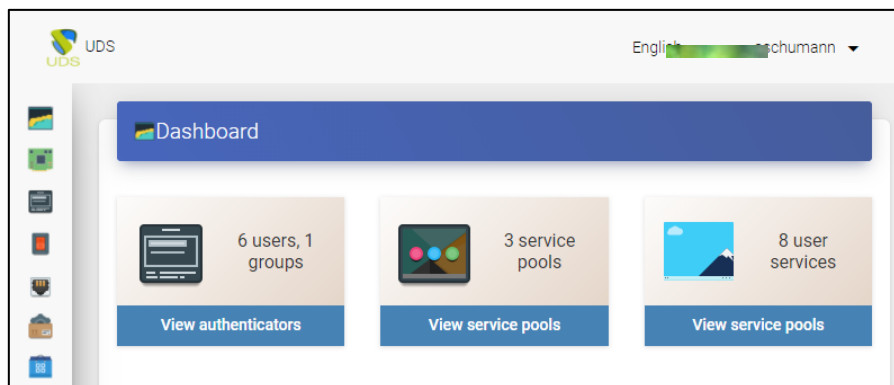

The first step to take will be to register the base service created in the virtual platform. In this example, it will be created in a VMware vSphere environment, so within the panel you will access the services side menu.
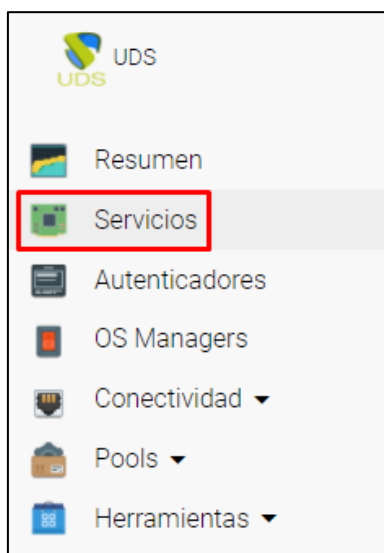

In this example, from the list of registered services, vCenter will be selected, which is where the Linux template is created (if you do not have a registered service provider, you will have to add one).
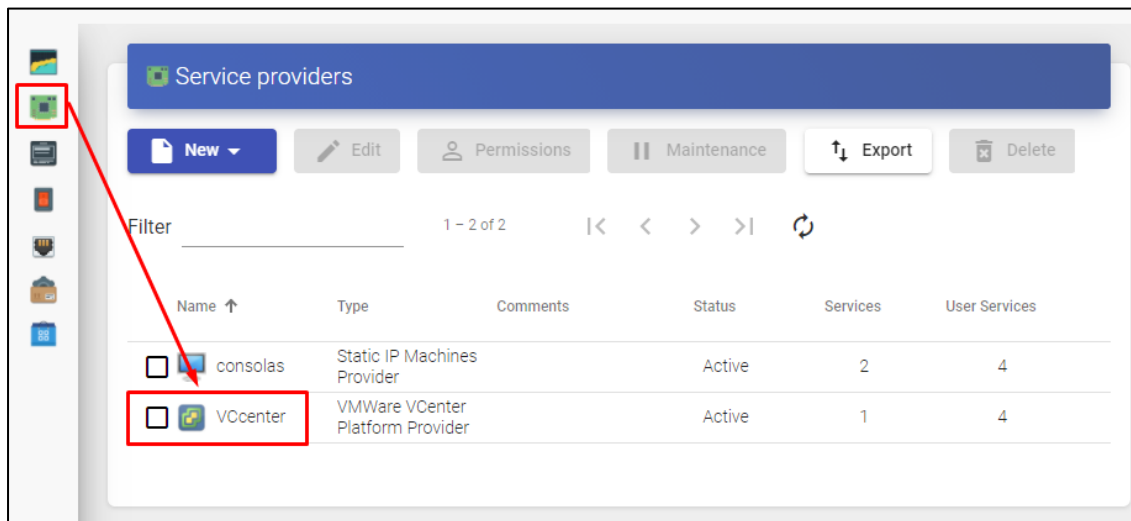
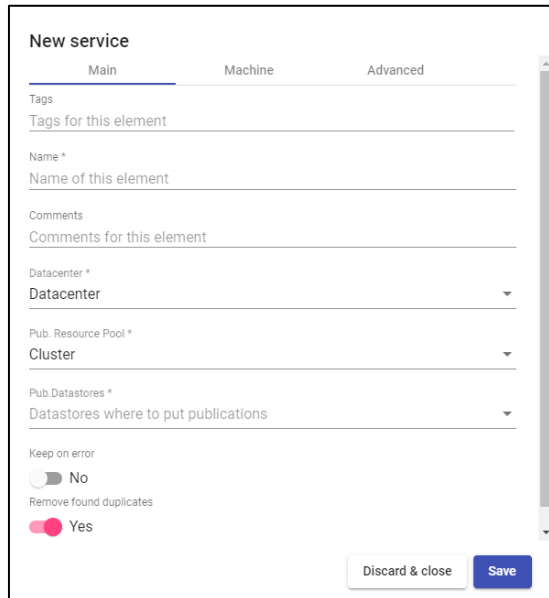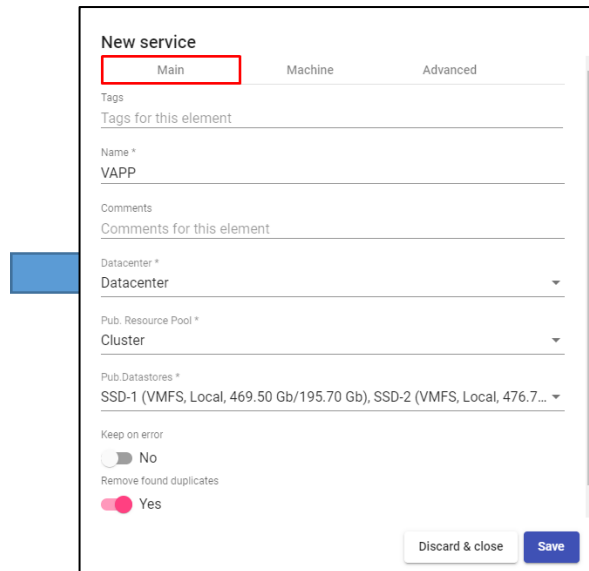It is selected and in the Services tab, click on the "New" button.
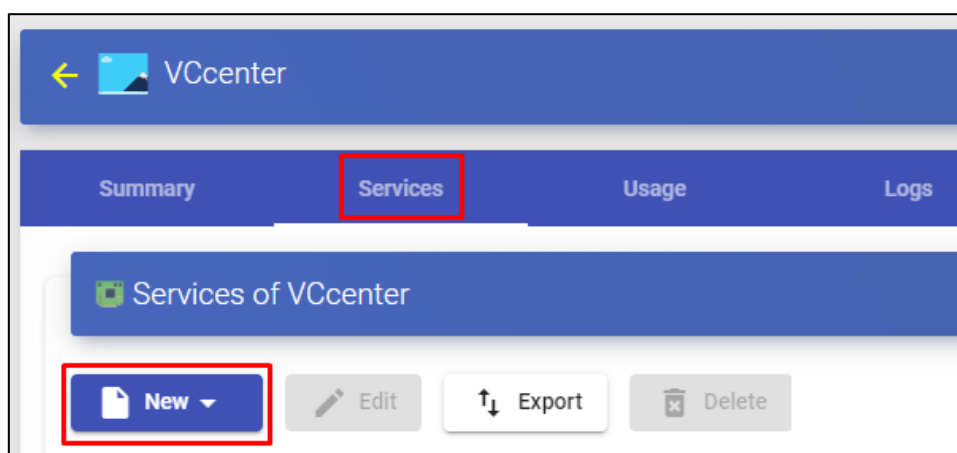
In the drop-down the linked clone type is selected (which will help to make very fast deployments and save resources).
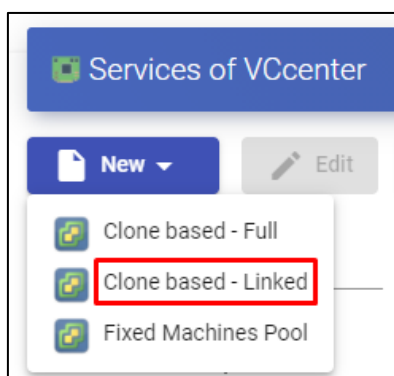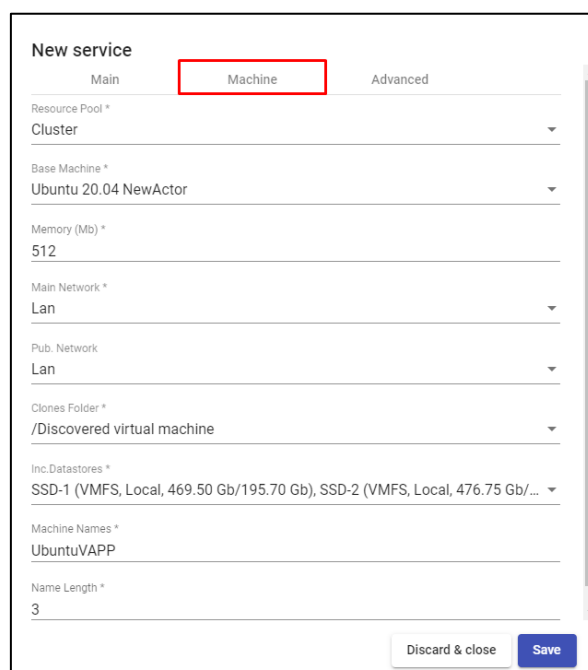
A pop-up window like the one below will appear.

For this type of provider (VMware vSphere) you will have to assign an identifying name, select the Datacenter, the resource pool where they will be published and in which pool the clones will be created.


In the "machine" tab, the Linux base machine (template) that has been created previously (Ubuntu) will be selected, as well as the memory that you will want to give to the auto-generated desktops, the network to which you want to connect them, the name DNS of the desktops, location and the length of the name. Once all this information has been completed, choose "Save".
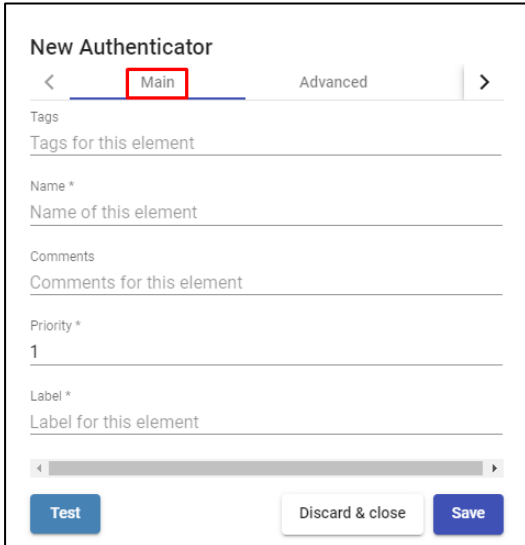
The next step is to register the authenticator that we are going to use in UDS Enterprise. In the case of not having an external one (AD, OpenLDAP, etc ...), we can use the internal database provided by UDS Enterprise. To do this, in the side menu we will select "Authenticators":
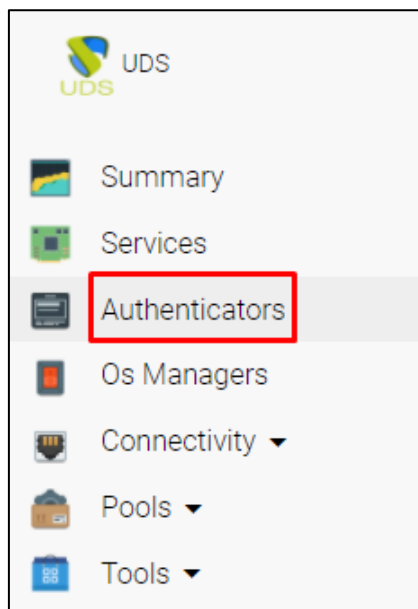
In our case, the internal database will be used. To register this database, click on the "new" button and select "internal database".
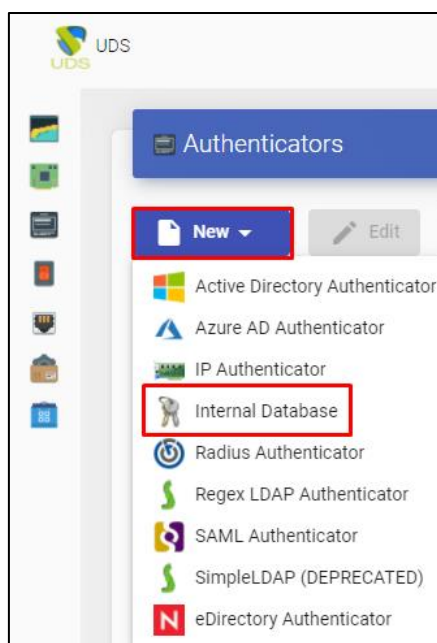
The following window will open in which you will have to add an explanatory name to later identify it on the platform, the priority of use and a tag name.
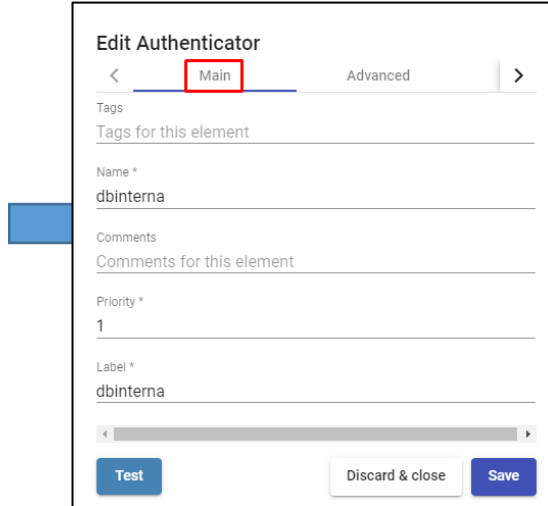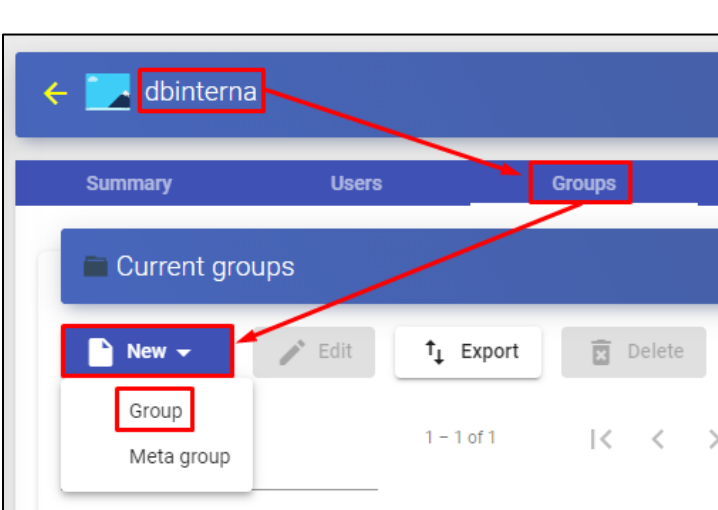
The next step will be to add the user groups that will be able to use the resource. To do this, you will have to select the database, go down in the window and access the "groups" tab.

The following window will open, in which the name of the group will be entered, click on the drop-down menu, leaving the status as active, and it will be saved.
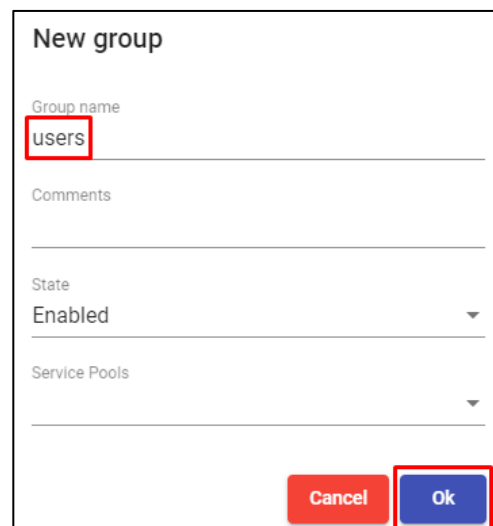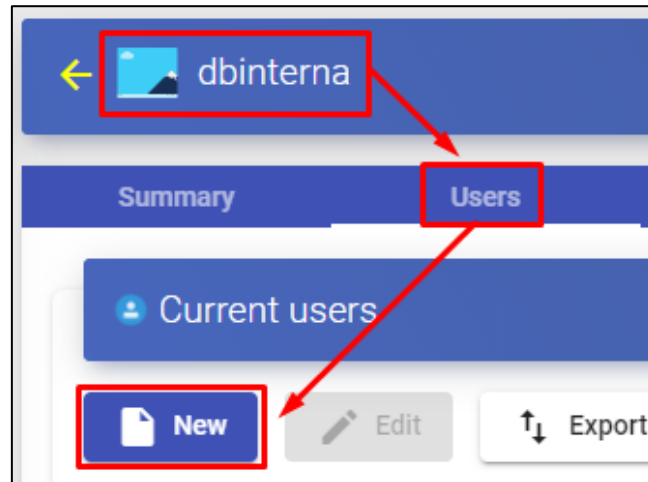
After creating the group, the users will have to be added. To do this, we enter the "users" tab and click on the "new" button.

The following window will appear, in which the username and password will be entered. The state will remain active. For this example you do not need to be a staff member or administrator, so these buttons will be left as "No", and the group created previously will be added. It is then saved.

The next step is to create the OS Manager. To do this, in the side menu choose "OS Manager" and select the "new" button.
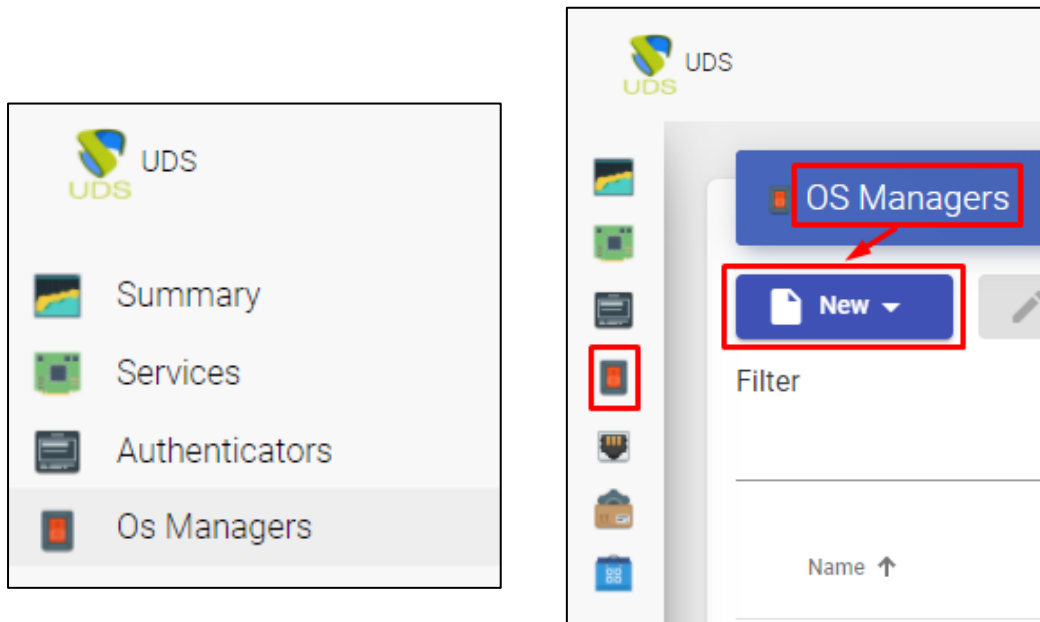
In the drop-down, select "Linux os Manager".
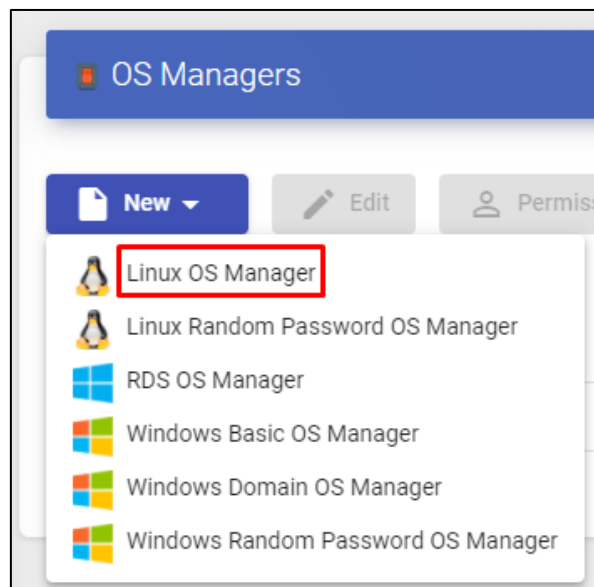
Enter a descriptive name and from the "Logout Action" drop-down select "Delete service (non-persistent desktops)". This option is recommended, since it allows that when a user closes the application, the desktop on which said application is running is automatically deleted (important for saving virtual platform resources). It is also possible to use persistent desktops to deploy Linux applications.
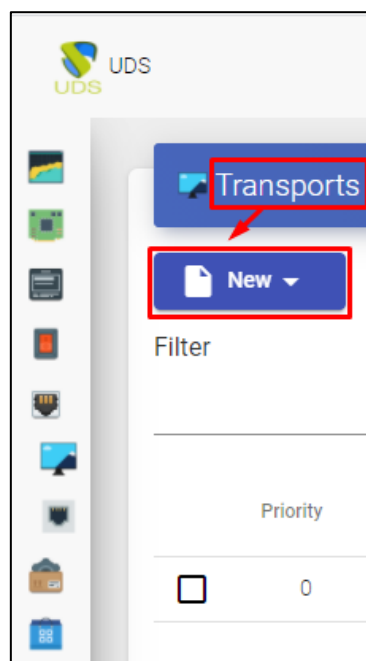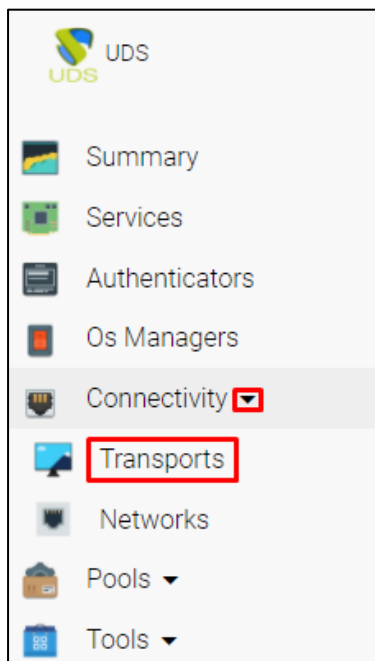
The next step will be to create the transport. This element is very important, since this is where we are going to tell UDS to use the autogenerated virtual desktops, based on the Linux template machine, to deploy Linux virtual applications. To create a new transport, the "Connectivity" tab will be selected in the side menu.
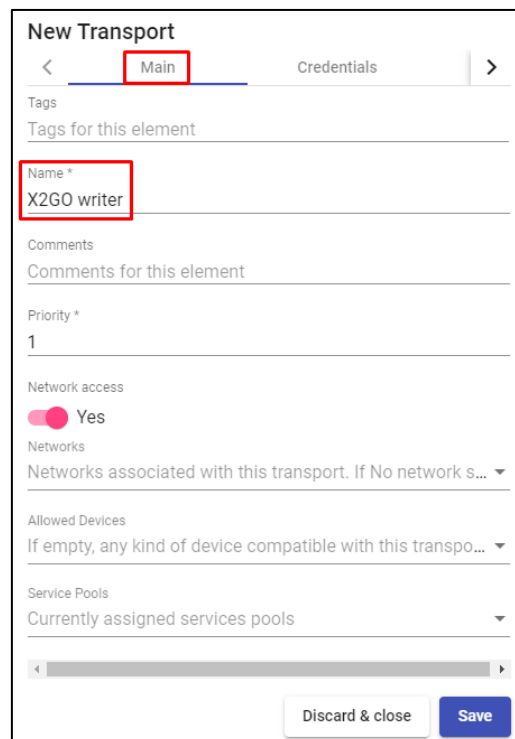
In the transport section, press the "new" button.

In this configuration example, the "X2Go direct" transport is selected from the drop-down menu, since it will only be used from the local network. In the event that the connection is made from outside the local network, the tunnelled X2Go protocol will have to be used.
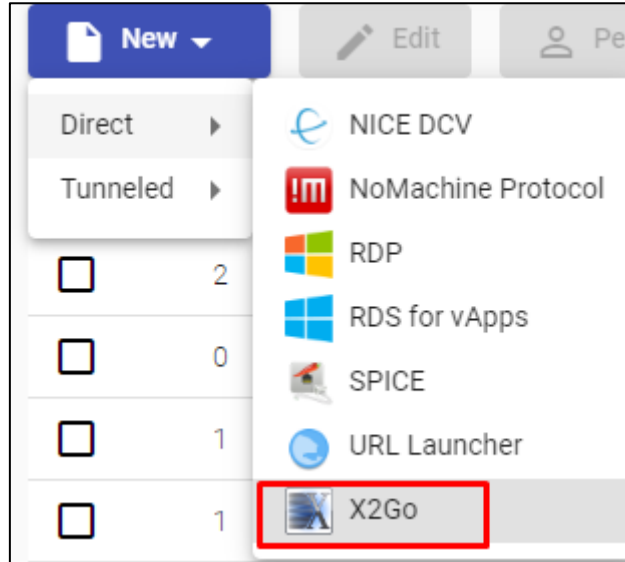
The following window will appear, in which you will have to add the identifying name (it is advisable to define the name of the application here, this name will be visible by the user only in the case that several transports are defined on the same resource pool) and if you want to limit access. In this case, it will be left by default, so that it can be accessible from any device and network.
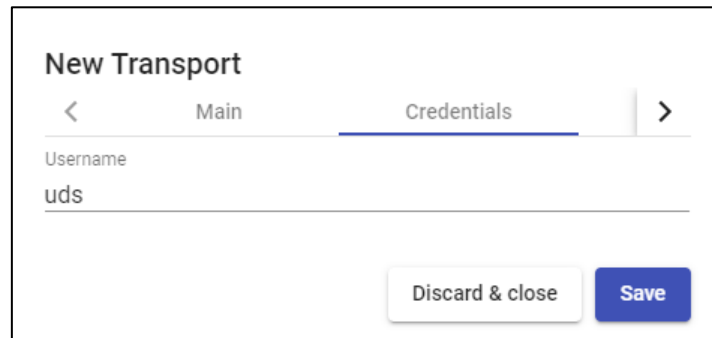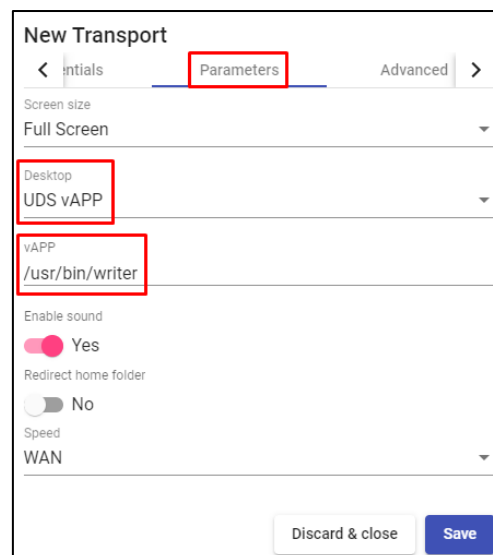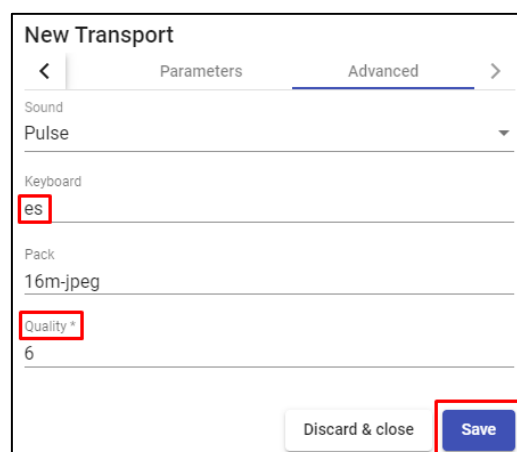
In the credentials tab, the user with whom the application will be run will be indicated. In this case, the user uds will be used (this user must exist in the Ubuntu template).

In the "parameters" tab, you will have to select UDS vAPP from the drop-down menu and indicate the full path of the application (installed on the Linux template machine) that you want to run.
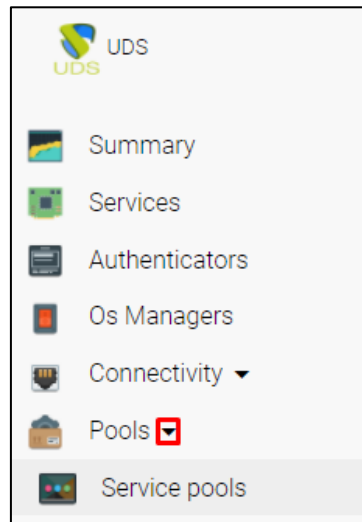

In the "advanced" tab, you enter the keyboard language and in "quality" you can define the image quality (0 - 9), being by default at 6.
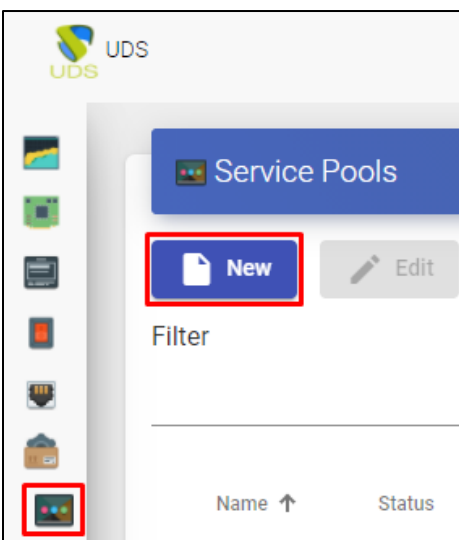
Once this configuration is done "save".

NOTE: You can create as many transports as applications need to be published for users.

Finally, the service pool will have to be created. To do this, go to the side menu to the "Service Pools" section.

Select "new":

The name of the service will be indicated (this name will be visible by the user and will help to identify the service), the base service and the OS Manager (previously created) are selected.

In the "screen" tab, it will be indicated in which group of services it is going to be displayed and which image the application is going to use (these elements are created in the "tools" section) and in the "advanced" tab the checks are left by default .

In the "availability" tab, you will have to indicate the number of services that we want to have available for the connection of users (it is advisable to have as many services, initial or in cache, as users will access the applications). Each user that connects will need a desktop and all the applications that we have defined in the transports will run on that desktop.

Once this configuration is made, it is saved so that the creation of the publication of the service and the creation of the virtual desktops in which the virtual applications will be delivered begins.
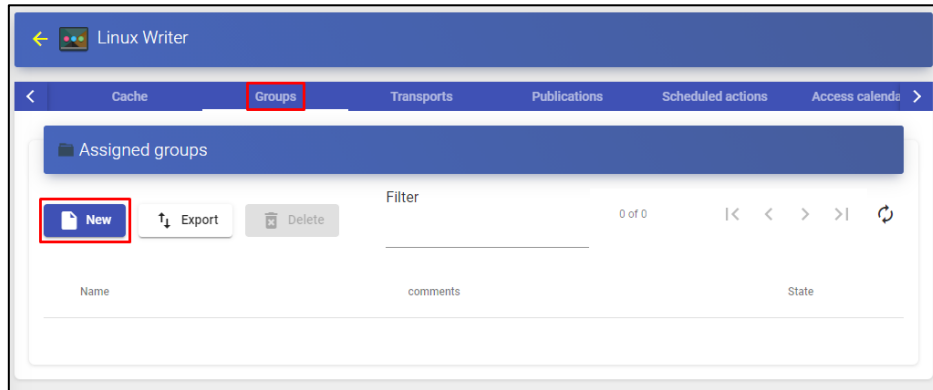
The next task will be to indicate which groups of users will access the virtual application service. Within the created service pool you will access the "groups" tab.
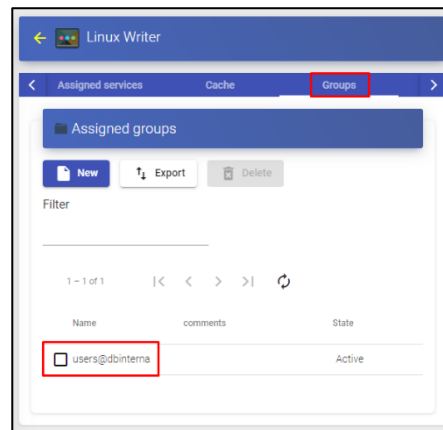

"New" is chosen and the authenticator / group (previously created) is selected and saved.
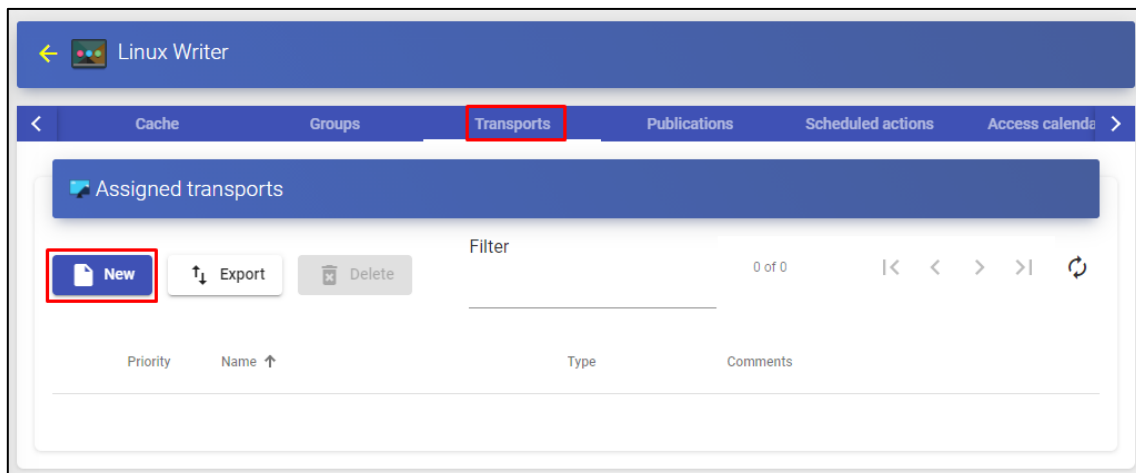

Once the group has been added, click on the "Transports" tab and assign the X2Go transport (where the path of the application to be virtualized has been defined) that has been previously created by clicking on "new".

Select the transport and click on "save". A single transport or more than one may be added per service pool.

Before connecting to an application, it should be verified that the UDS auto-generated desktops have been configured correctly. To do this, it will be checked in the "Cache" tab that we have the services in a valid state.

Access to Linux application services

Once all these configurations have been made, and verified that the Linux desktops have auto-generated correctly and are in a valid state, users will be able to access the virtual application service.
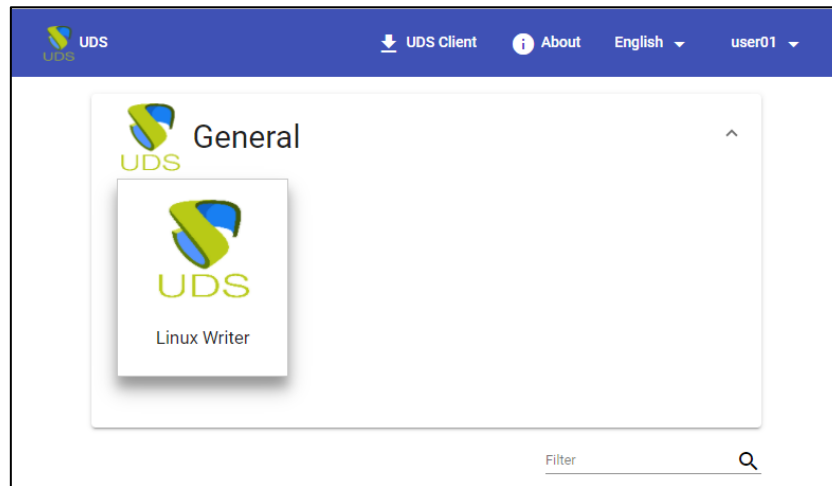
When the user accesses (he must have the X2Go client installed and the UDS client, both for Windows and Linux platforms), the following icon will appear on the taskbar:
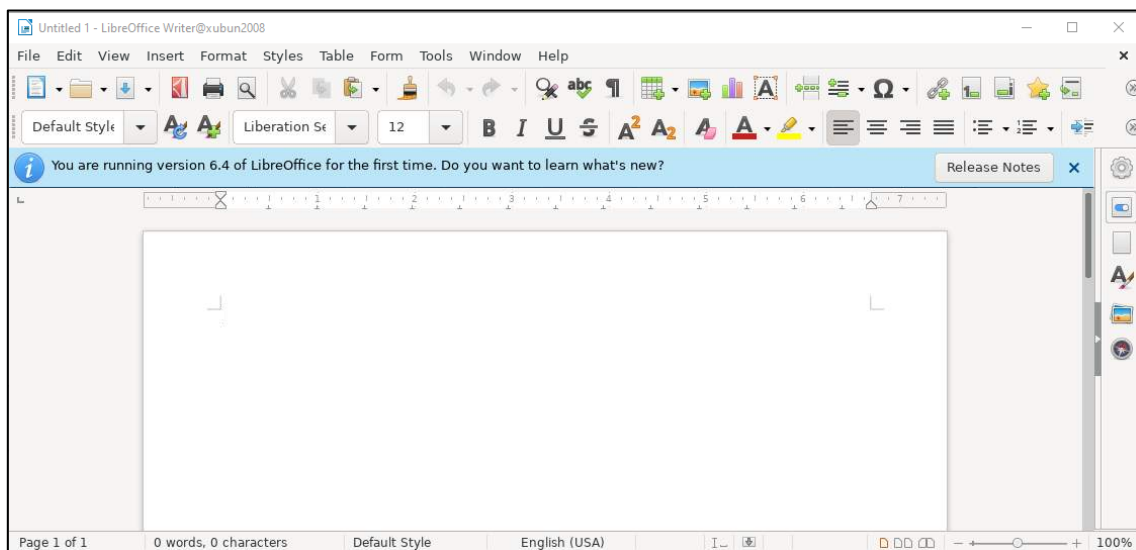

And the application will open.

In the event that more than one application per pool of services is chosen, you can select the application to which you want to connect by clicking on the gear in the image and in the list click on the application you want to run:

About Virtual Cable

Virtual Cable is a company specialized in the **digital transformation** of the **workplace**. The company develops, supports and markets UDS Enterprise. Its team of experts has designed **VDI** solutions tailored to **each sector** to provide a unique user experience fully adapted to the needs of each user profile. Virtual Cable professionals have more than **30 years of experience** in IT and software development and more than 15 in virtualization technologies. **Millions of Windows and Linux virtual desktops with UDS Enterprise are deployed all over the world every day**.