VIRTUAL CABLE

Linux Application Virtualization UDS Enterprise 4.0

#SmartDigitalWorkplace

VIRTUAL CABLE

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 may be assigned to groups of users.

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

Users will run the applications on 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 these applications will run on the same virtual desktop, thus achieving a significant optimization of resources.

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

Required Items

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

1. Virtualization Platform

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

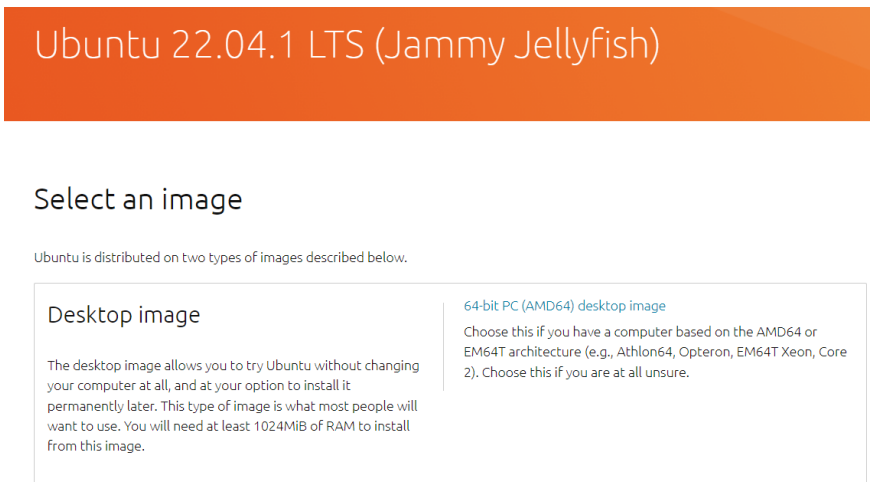
The virtualization platforms supported by UDS Enterprise to be able to deploy Linux applications are: Citrix XenSenver / XCP-ng, Microsoft Azure, Microsoft HyperV, oVirt/RHEV, VMware vSphere, VMware vCloud, Nodeweaver, OpenNebula, OpenStack, etc...

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 users will run their virtual applications.

It is recommended to use an Ubuntu distribution as a template. In this example, we are going to use a minimum image of Ubuntu 22.04, available from the official Ubuntu repository: https://ubuntu.com/download/desktop

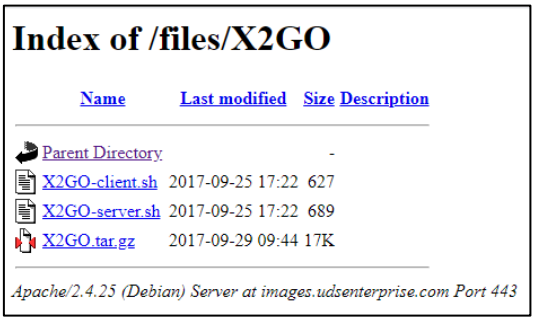


3. X2Go Protocol

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

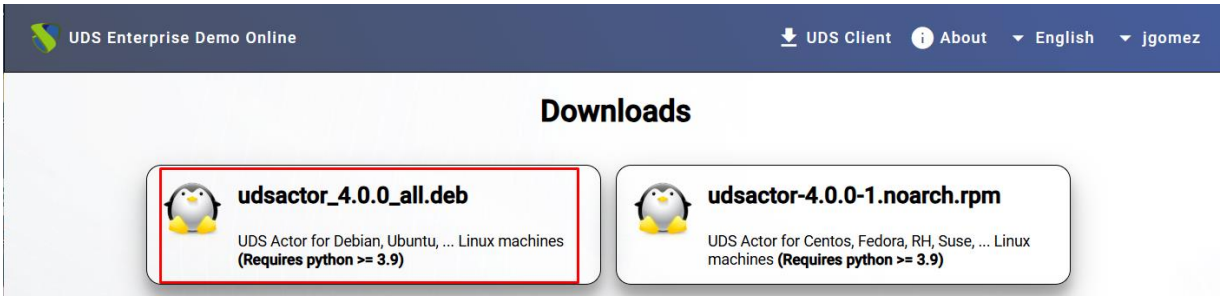
The UDS Enterprise team has created a script that automates and simplifies the entire X2Go installation process. This script (we have generated one for the server part, which we installed in the template, and another for the client part of Linux computers, which is installed in the connection client)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

http://images.udsenterprise.com/files/X2GO/



4. UDS Actor

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



5. Miscellaneous

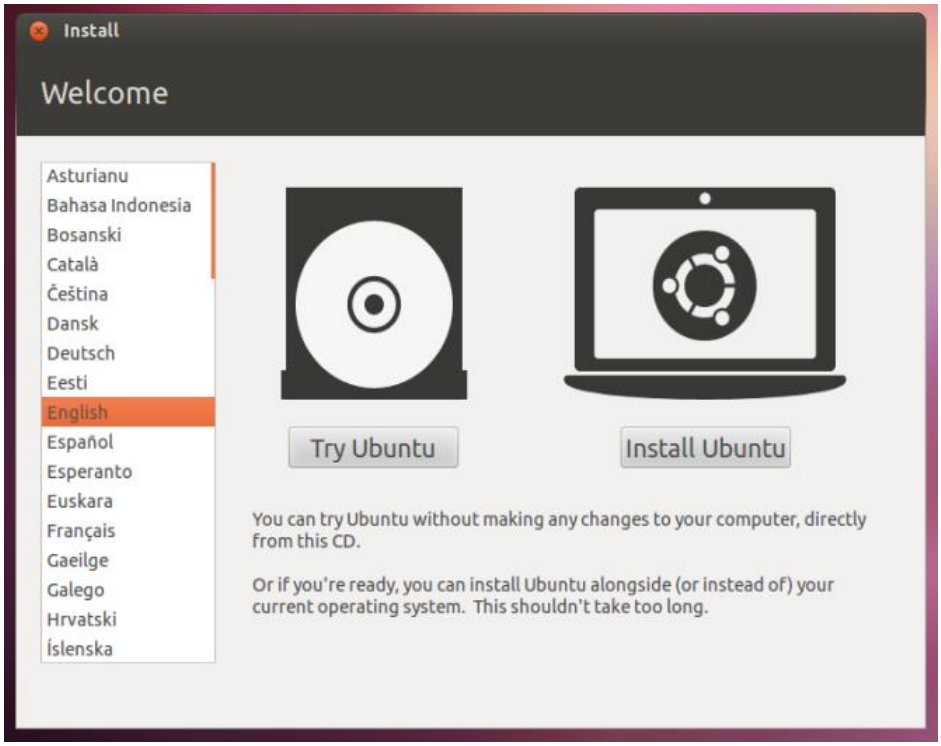
It is necessary to have an Internet output on the template machine to be able to install both the Ubuntu OS and the X2Go connection protocol. It will also be necessary to have a DHCP server on the network where the virtual desktops are generated.

Configuring the Linux Template and Connection Client

Below are all the necessary tasks to be performed on both the Linux base machine (template) and the connection client (computer that will access the Linux virtual applications).

1. Linux OS

You will start by creating the VM that is going to be used as a base machine (template). Once you have downloaded the Ubuntu 22.04 image from the official repository indicated in the previous section, you can proceed to install:

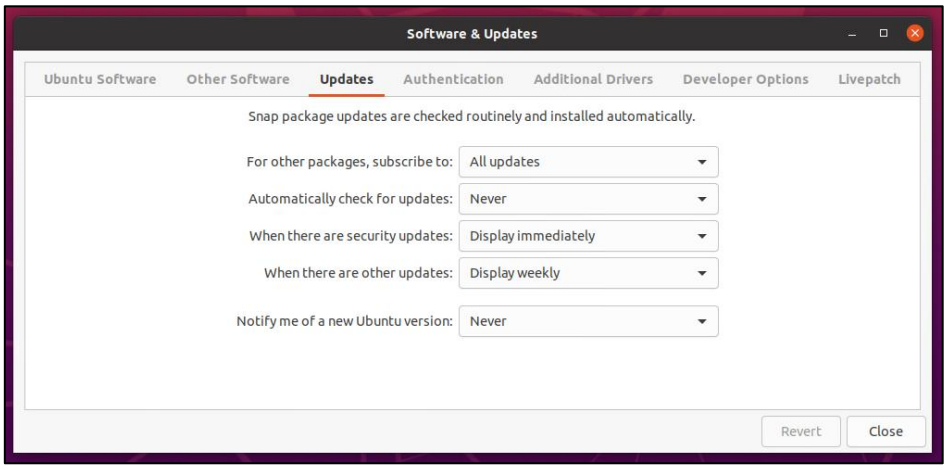


The installation will be carried out with the default values and indicating resources (disk, vRAM and vCPUs) according to the requirements of the applications that will be installed on the machine.

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 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, to do it manually.



2. X2Go Protocol – Server

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

```
sudo apt-get update
```

```
uds@vc-virtual-machine:~$ sudo apt-get update
```

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

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

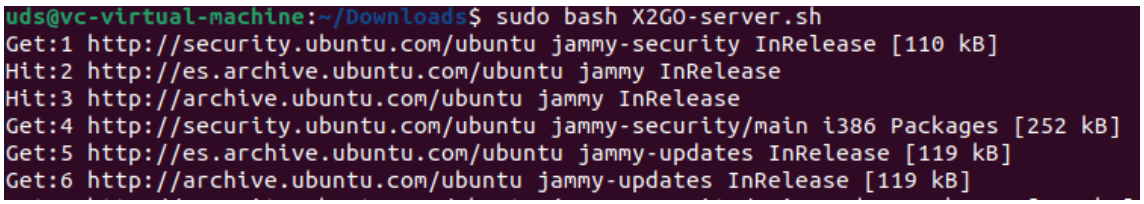
```
uds@vc-virtual-machine:~$ wget http://images.udsenterprise.com/files/X2GO/X2GO-server.sh
--2023-02-09 09:38:17--  http://images.udsenterprise.com/files/X2GO/X2GO-server.sh
Resolving images.udsenterprise.com (images.udsenterprise.com)... 188.165.133.128
Connecting to images.udsenterprise.com (images.udsenterprise.com)|188.165.133.128|:80... c
onnected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://images.udsenterprise.com/files/X2GO/X2GO-server.sh [following]
--2023-02-09 09:38:17--  https://images.udsenterprise.com/files/X2GO/X2GO-server.sh
Connecting to images.udsenterprise.com (images.udsenterprise.com)|188.165.133.128|:443...
connected.
HTTP request sent, awaiting response... 200 OK
Length: 689 [text/x-sh]
Saving to: 'X2GO-server.sh'

X2GO-server.sh      100%[==================>]     689  --.-KB/s    in 0s

2023-02-09 09:38:17 (109 MB/s) - 'X2GO-server.sh' saved [689/689]

uds@vc-virtual-machine:~$
```

We locate where the file has been downloaded.

```
uds@vc-virtual-machine:~/Downloads$ ls
X2GO-server.sh
uds@vc-virtual-machine:~/Downloads$
```

The file is executed:

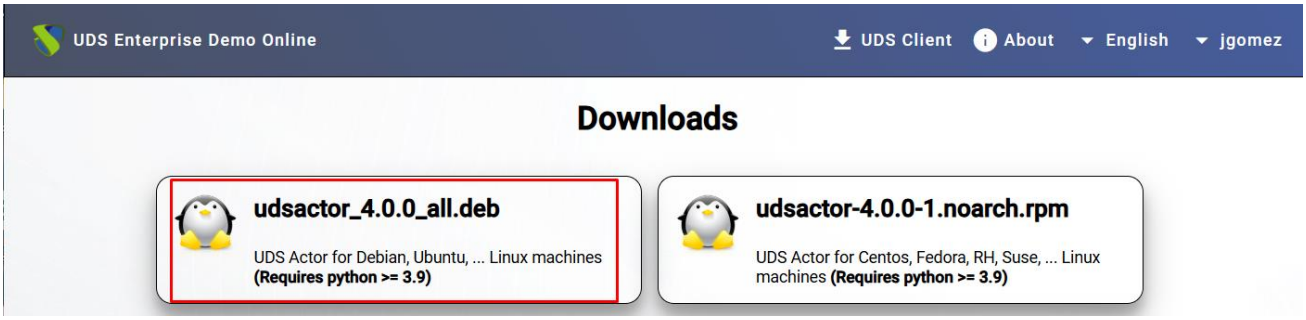
```
Sudo sh /home/uds/X2GO-server.sh
```



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

3. Actor UDS en Linux

The UDS Actor for Debian-based distributions is downloaded.



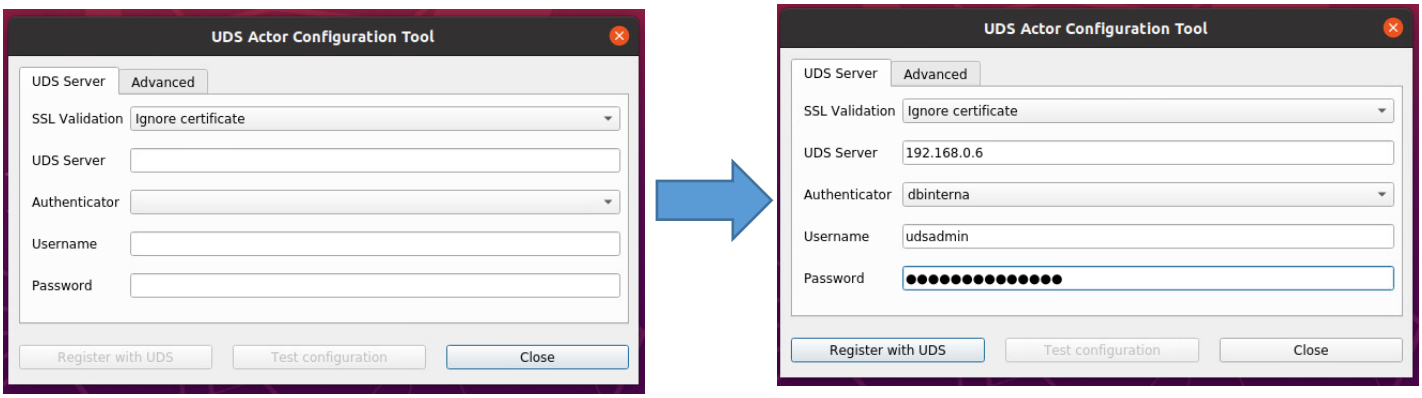
For the installation of the UDS Actor it will be necessary to have version 3.9 or higher of Python.

```
uds@vc-virtual-machine:~/Downloads$ sudo apt install python3
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python3 is already the newest version (3.10.6-1~22.04).
python3 set to manually installed.
The following packages were automatically installed and are no longer required:
  libreoffice-ogltrans linux-headers-5.15.0-43 linux-headers-5.15.0-43-generic
  linux-image-5.15.0-43-generic linux-modules-5.15.0-43-generic
  linux-modules-extra-5.15.0-43-generic
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 245 not upgraded.
uds@vc-virtual-machine:~/Downloads$
```

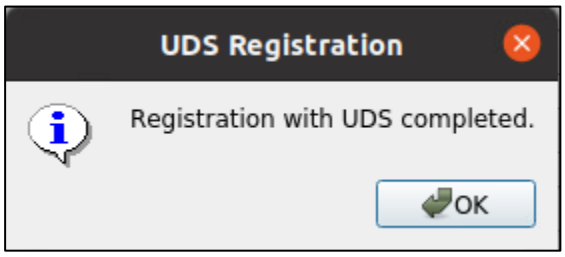
The installation of the .deb package containing the UDS Actor is carried out:

```
uds@vc-virtual-machine:~/Descargas$ sudo dpkg -i udsactor_4.0.0_all.deb
[sudo] password for uds:
(Reading database ... 213597 files and directories currently installed.)
Preparing to unpack udsactor_4.0.0_all.deb ...
Unpacking udsactor (4.0.0) over (4.0.0) ...
Setting up udsactor (4.0.0) ...
Processing triggers for mailcap (3.70+nmu1ubuntu1) ...
Processing triggers for gnome-menus (3.36.0-1ubuntu3) ...
Processing triggers for desktop-file-utils (0.26-1ubuntu3) ...
uds@vc-virtual-machine:~/Descargas$
```

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



In case you need to modify any parameter, you can do it from this window or also by editing the file: /etc/udsactor/udsactor.cfg




```
uds@vc-virtual-machine:~$ sudo su
[sudo] password for uds:
root@vc-virtual-machine:/home/uds#  vi /etc/udsactor/udsactor.cfg
```

```
[uds]
host = 192.168.0.6
validate = no
type = managed
master_token = A2LXBxcq7LA33ReVhIP--PheFdbNrqtp6oHJsebrQKgFEhJ3
log_level = 2
```

4. X2Go Protocol – Connection Client

LINUX

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

```
sudo apt-get update
```

```
uds@vc-virtual-machine:~$ sudo apt-get update
```

Once updated, X2Go can be installed on the client through the script provided by VirtualCable (it is necessary to run the script with a graphical environment session running). You will have to copy the script in your Linux client:

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

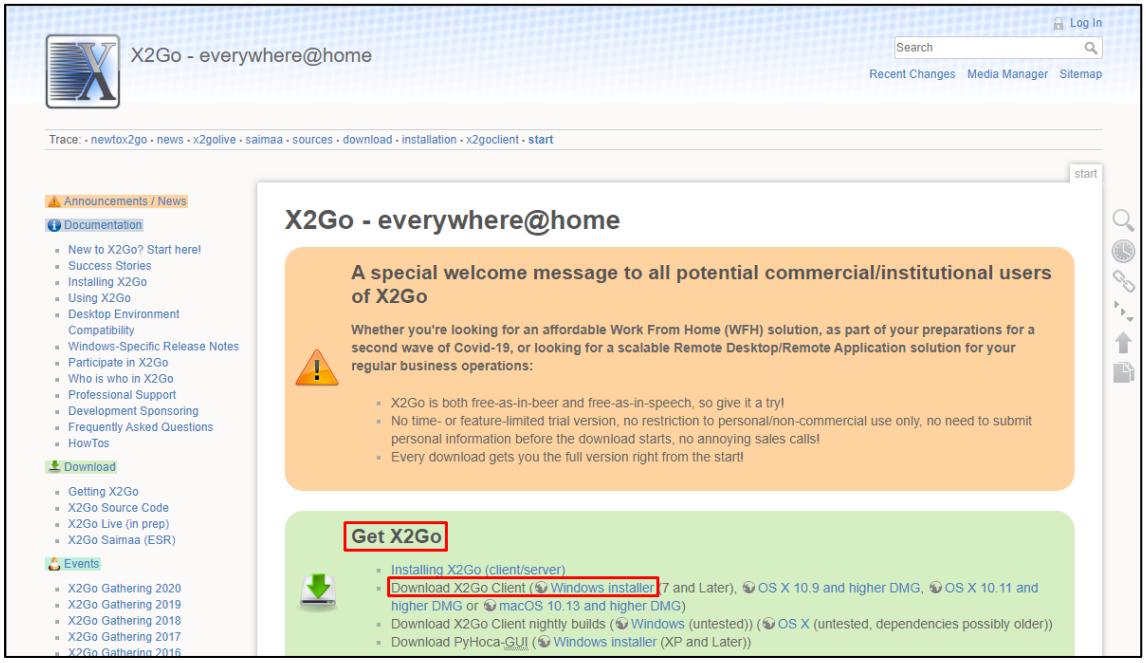
```
uds@vc-virtual-machine:~$ wget http://images.udsenterprise.com/files/X2GO/X2GO-client.sh
--2023-02-09 09:49:02--  http://images.udsenterprise.com/files/X2GO/X2GO-client.sh
Resolving images.udsenterprise.com (images.udsenterprise.com)... 188.165.133.128
Connecting to images.udsenterprise.com (images.udsenterprise.com)|188.165.133.128|:80...
ed.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://images.udsenterprise.com/files/X2GO/X2GO-client.sh [following]
--2023-02-09 09:49:02--  https://images.udsenterprise.com/files/X2GO/X2GO-client.sh
Connecting to images.udsenterprise.com (images.udsenterprise.com)|188.165.133.128|:443...
ted.
HTTP request sent, awaiting response... 200 OK
Length: 627 [text/x-sh]
Saving to: 'X2GO-client.sh'

X2GO-client.sh          100%[=============================>]     627  --.-KB/s    in 0s

2023-02-09 09:49:02 (421 MB/s) - 'X2GO-client.sh' saved [627/627]

uds@vc-virtual-machine:~$
```

The script is located and executed:

```
uds@vc-virtual-machine:~$ sudo sh X2GO-client.sh
[sudo] password for uds:
Hit:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
```

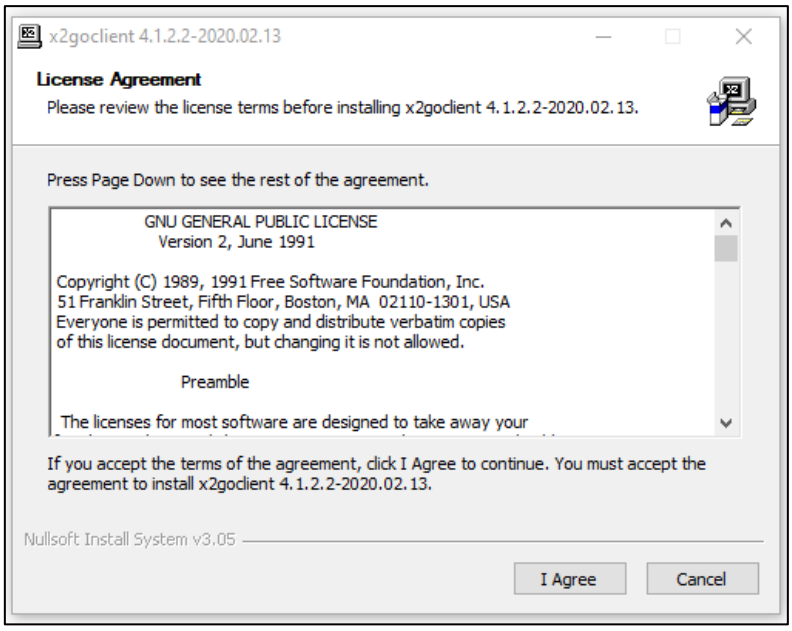
WINDOWS

If you have a Windows client, you will have to download the X2GO client from the following link: https://wiki.X2Go.org/doku.php

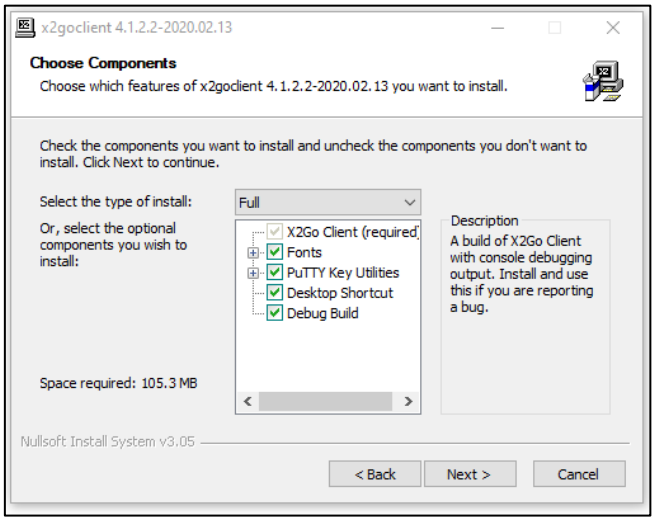


After finishing the download, the X2Go client installer is executed, following the appropriate steps for each situation:

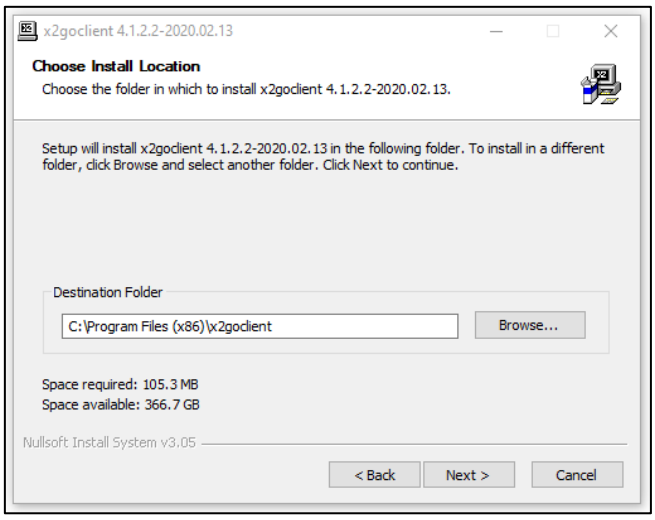
You will have to accept the terms of the license of use by clicking on agreement:



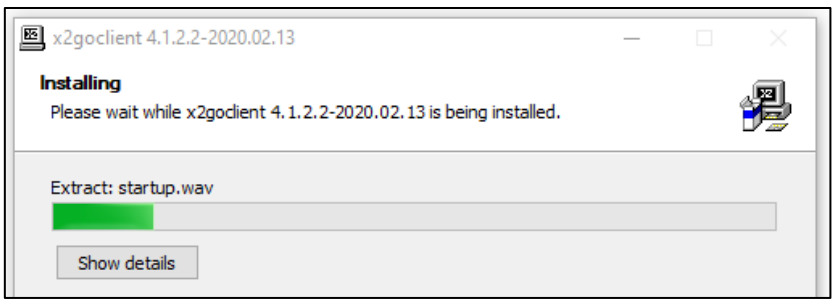
The components to be installed are selected. In this case, select the type of installation in "Full" and click on "Next":



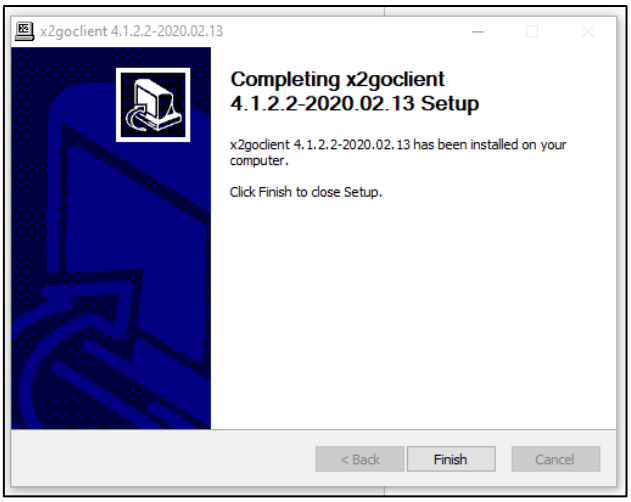
The path in which you want the installation to be carried out will be selected and select "Next":



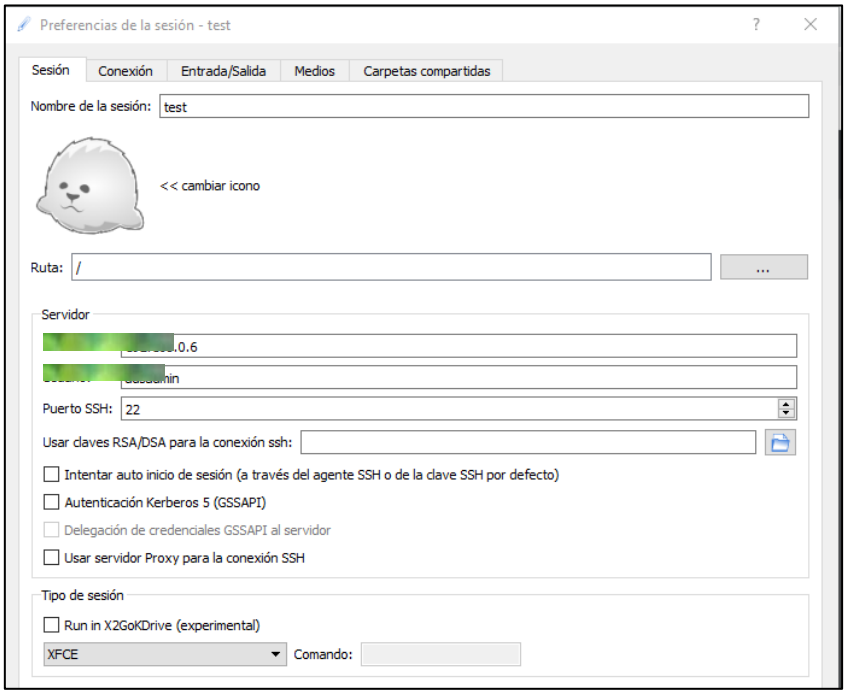
Select the folder in the start menu where you want to install:



After the installation is complete, it will indicate that it has been successful:



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

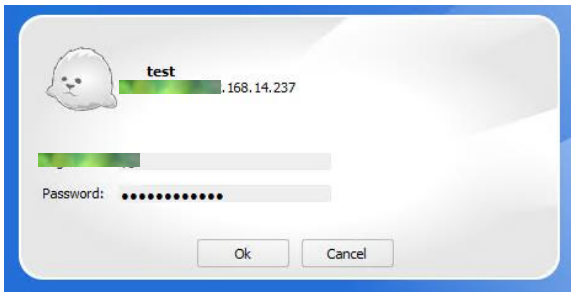


In the "**Host**" field, you enter the IP of the computer you want to connect to, in this case the template.

In the "**Login**" field, enter the username. XFCE must be selected as the "**Session type**".

You proceed to the login page and enter the password.

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

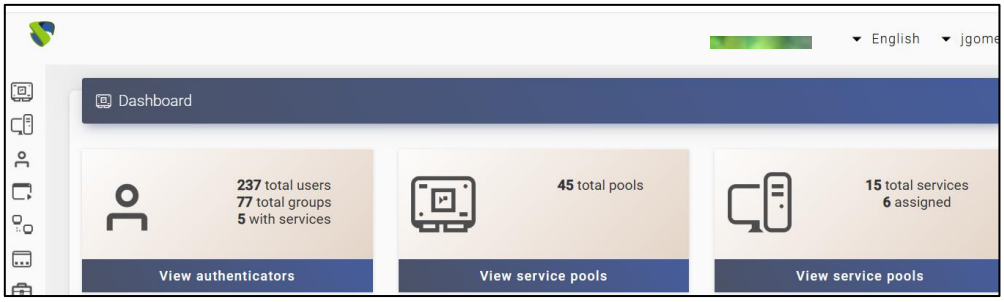


NOTE: On connection client computers, in addition to installing the X2Go client, it will also be necessary to have the UDS client (UDS Plugin) installed. For more details on its installation, review the guide: UDS Enterprise Installation, Administration and User Manual available in the documentation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

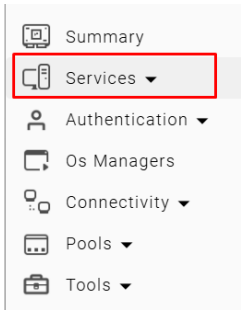
Service Publication

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

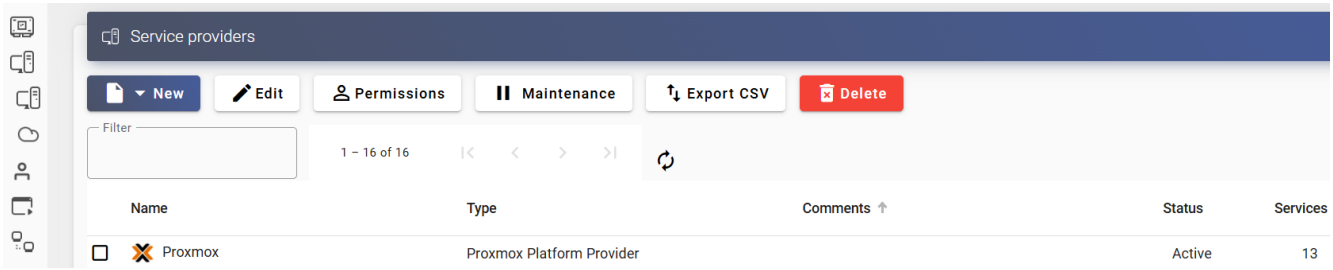
In order to configure a pool of services and deploy Linux applications, the UDS control panel will be accessed with an administrator user.



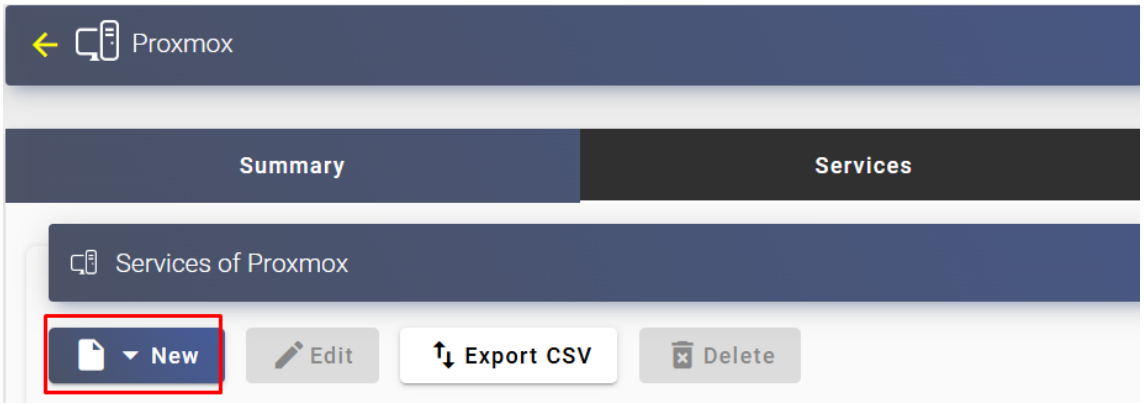
The first step to be taken will be to register the base service created on the virtual platform. In this example, it will be created in a VMware vSphere environment, so within the panel you will access the Services side menu.



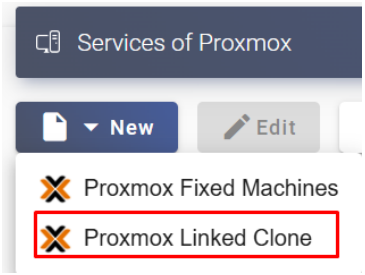
In this example, from the list of services that have been registered, vCenter will be selected, which is where the Linux template is created (if you do not have a service provider registered, you will have to add one).



Select and in the Services tab click on the "New" button.

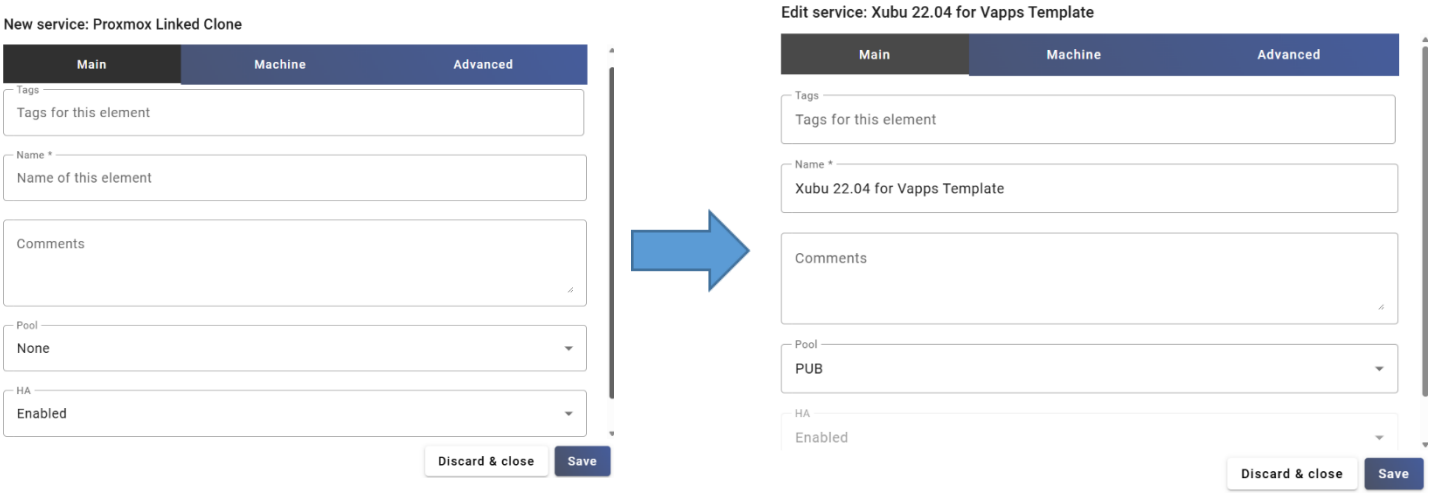


In the drop-down menu, the linked clone type is selected (which will help to make very fast deployments and save resources).

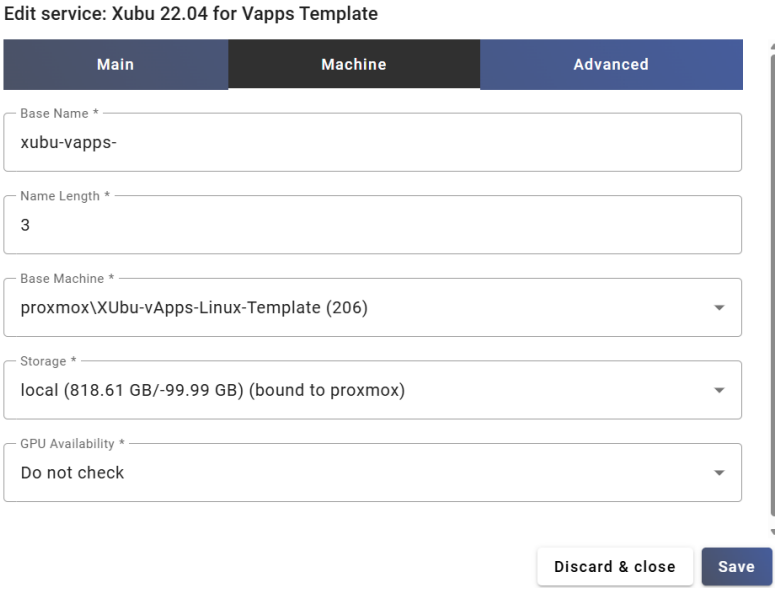


A pop-up window like the following will appear.

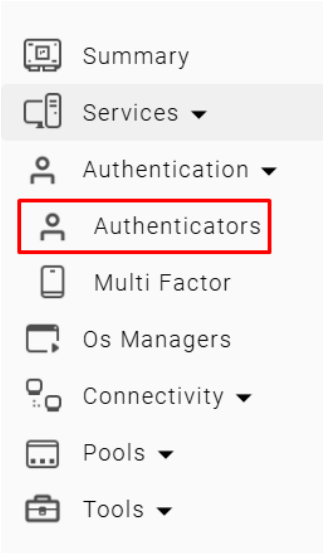
For this type of provider, you will have to assign an identifying name, the pool of resources where they are going to be published and where the clones are going to be created.



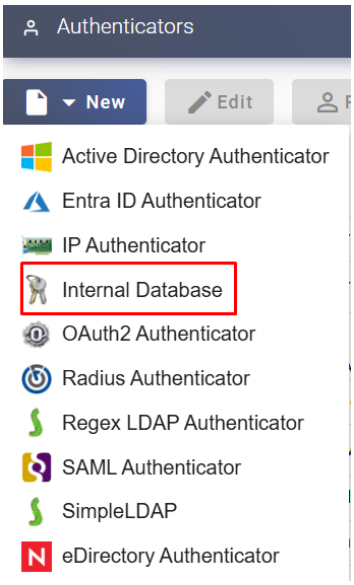
In the "machine" tab you will select the Linux base machine (template) that has been created previously (Ubuntu), the DNS name of the desktops, location and the length of the name. Once all this information has been completed, "Save" is chosen.



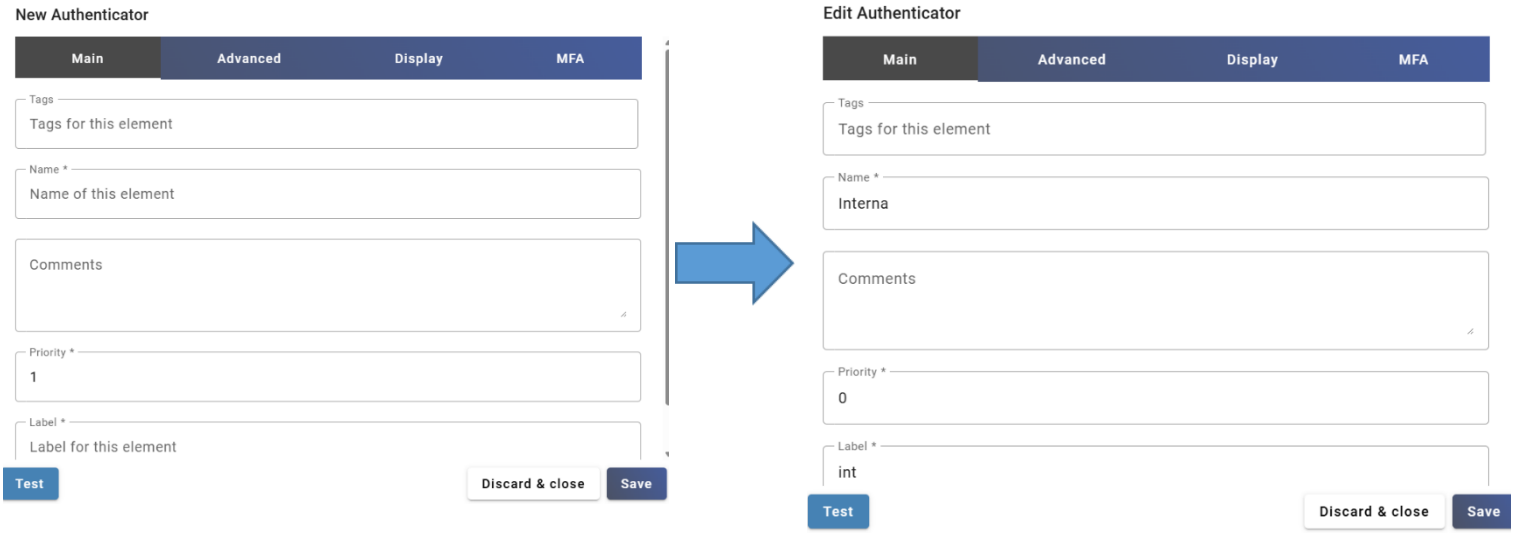
The next step is to register the authenticator that we are going to use in UDS Enterprise. In the case of not having an external one (AD, OpenLDAP, etc...), we can use the internal database provided by UDS Enterprise. To do this, in the side menu we will select "Authenticators":



In our case, the internal database will be used. To register this database, click on the "new" button and select "internal database".

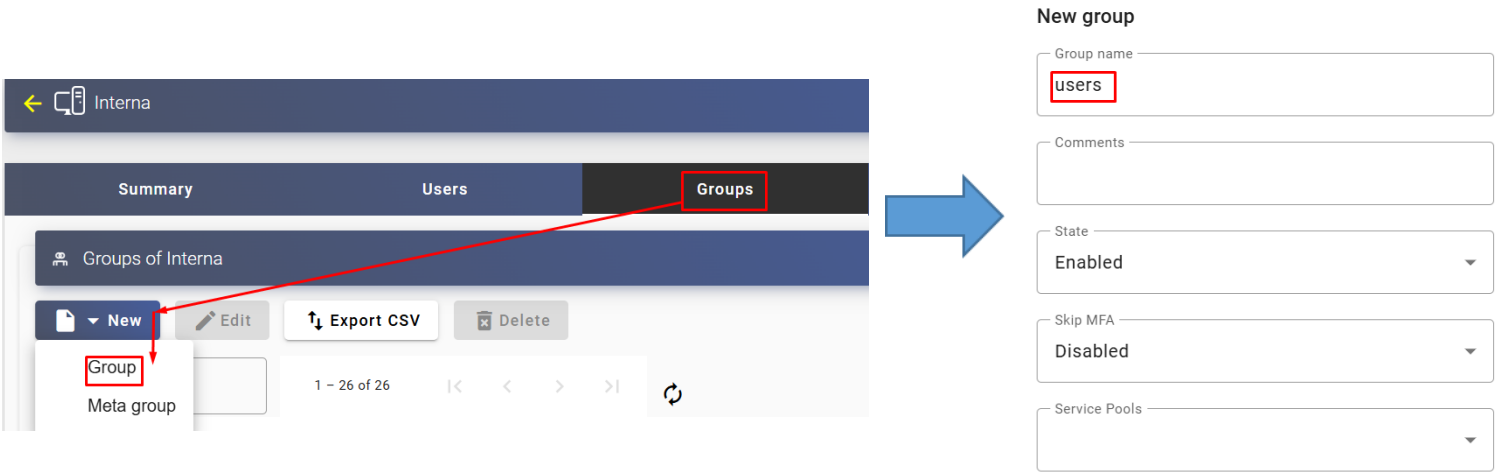


The next window will open in which you will have to add an explanatory name to then identify it on the platform, the priority of use and a tag name.

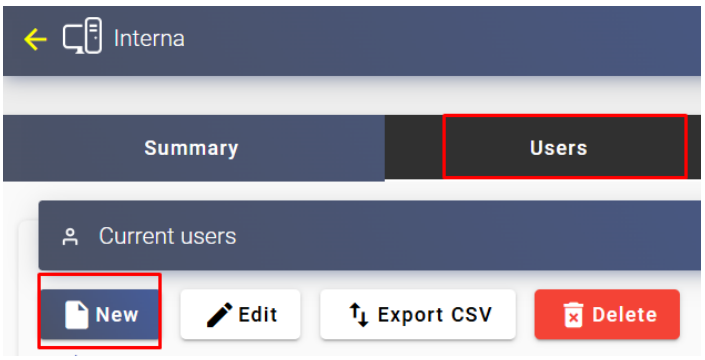


The next step will be to add the user groups that will be able to use the resource. To do this, you will have to select the database, scroll down in the window and access the "groups" tab.

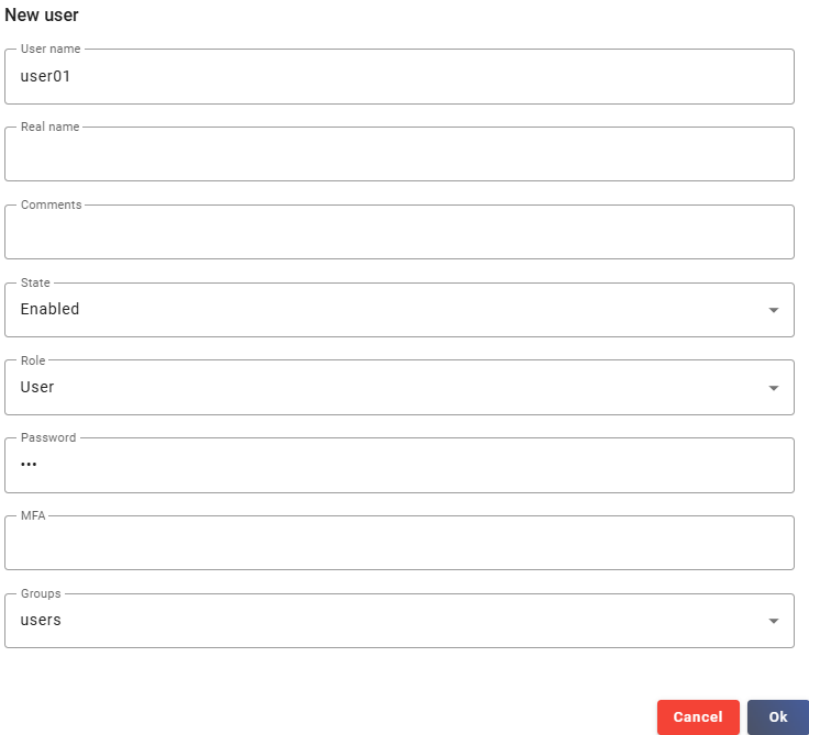
The following window will open, in which the name of the group will be entered, click on the drop-down drop-down leaving the status as active and it will be saved.



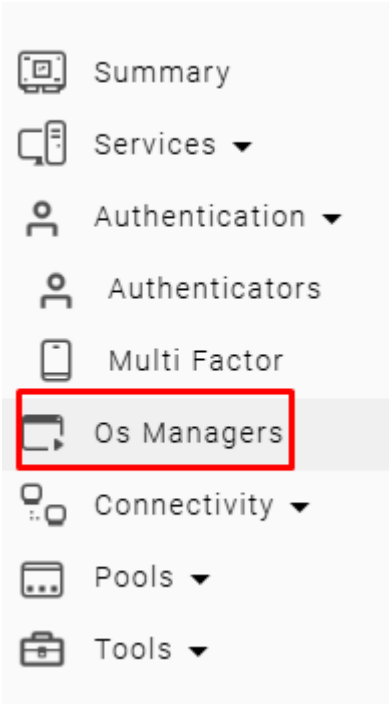
After creating the group, the users will have to be added. To do this, go to the "users" tab and click on the "new" button.

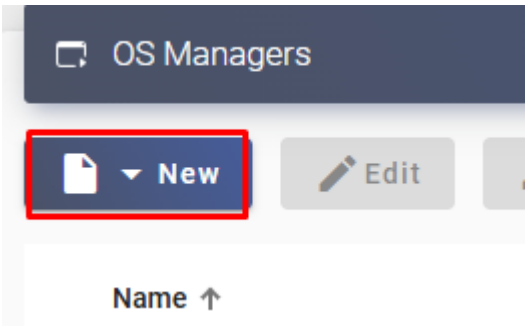


The following window will appear, in which the user's name and password will be entered. The status will remain as an asset. For this example, you don't need to be a staff member or administrator, so those buttons will be left as "No", and the previously created group will be added. It is then saved.

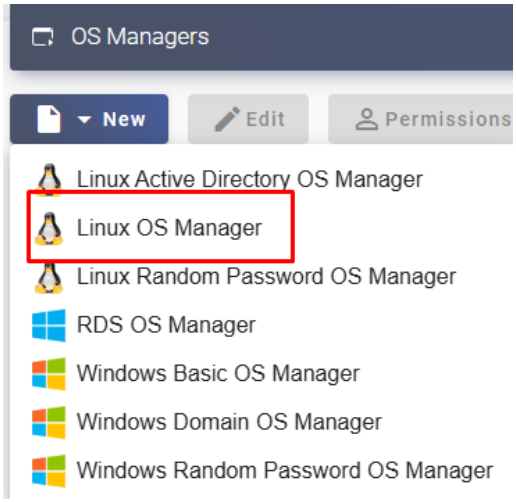


The next step is to create the OS Manager. To do this, in the side menu you will choose "OS Manager" and select the "new" button.

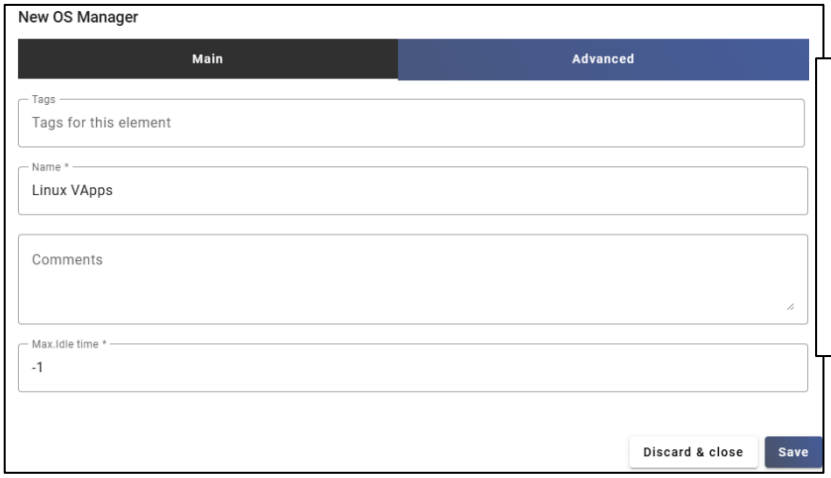


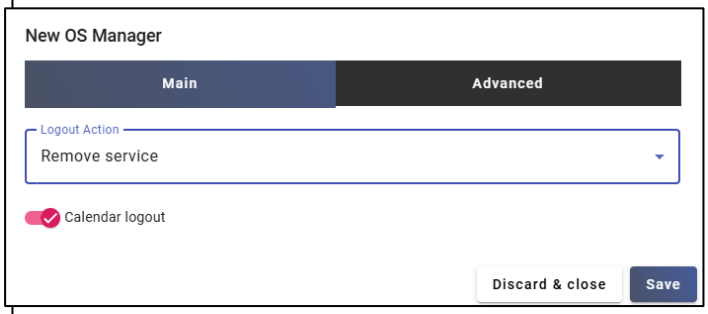


In the drop-down menu, select "Linux OS Manager".



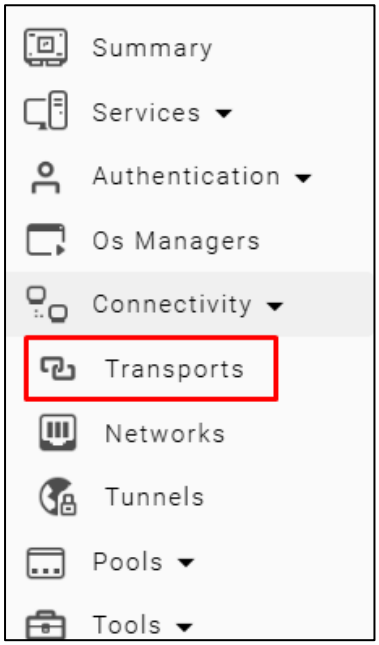
A descriptive name is entered and in the Advanced section under "Logout Action" "Delete service (non-persistent desktops)" is selected. This option is recommended, since it allows that when a user closes the application, the desktop on which the application is running is automatically deleted (important for saving resources of the virtual platform). It is also possible to use persistent desktops to deploy Linux applications.

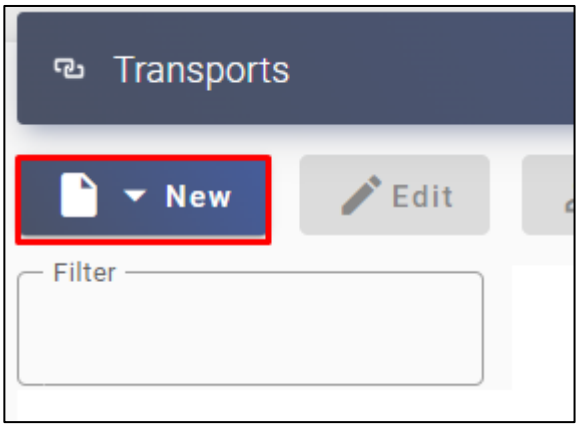




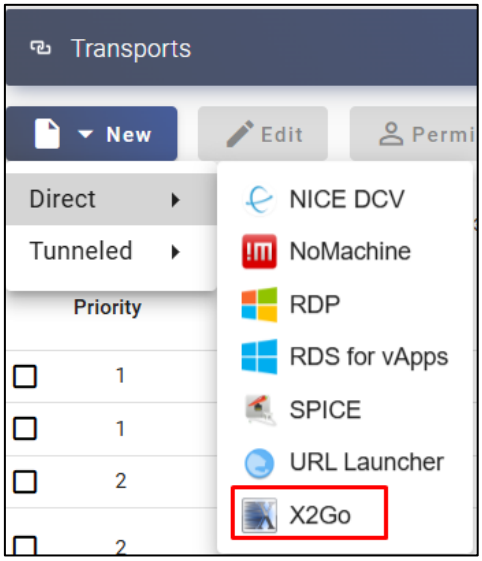
The next step will be to create the transport. This element is very important, since this is where we are going to tell UDS to use auto-generated virtual desktops, based on the Linux template machine, to deploy Linux virtual applications. To create a new transport, the "Connectivity" tab will be selected from the side menu.

In the transport section, the "new" button will be clicked.

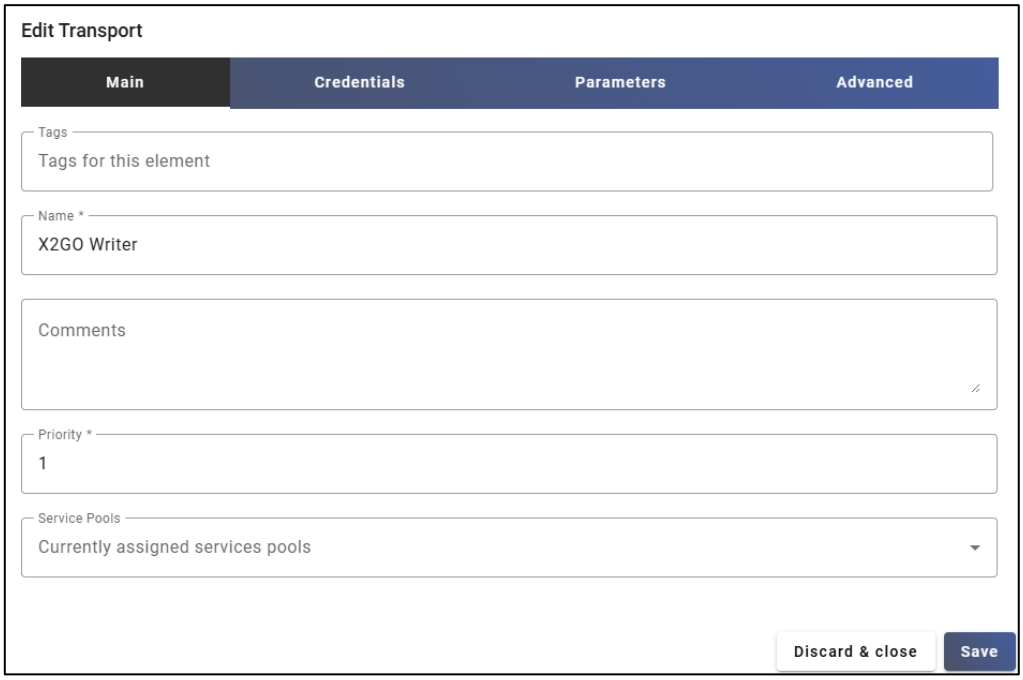




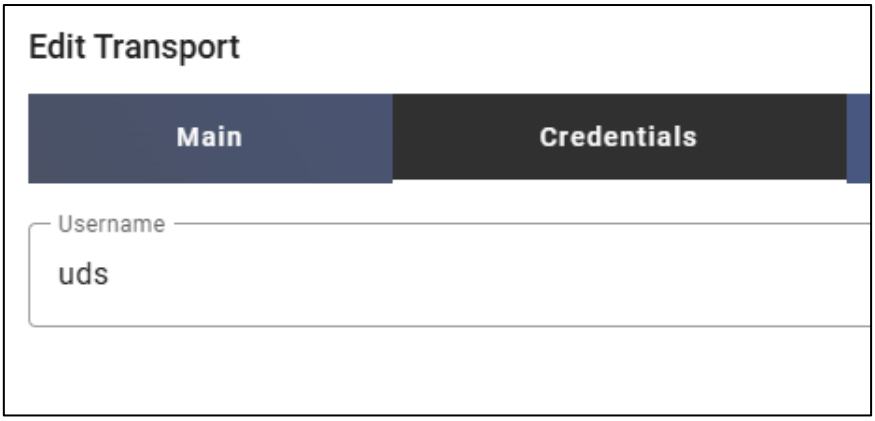
In this configuration example, the "X2Go direct" transport is selected from the drop-down menu, as it will only be used from the local network. In the event that the connection is made from outside the local network, the X2Go tunneled protocol will have to be used.



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 to the user only in the case that several transports are defined on the same pool of resources) 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.

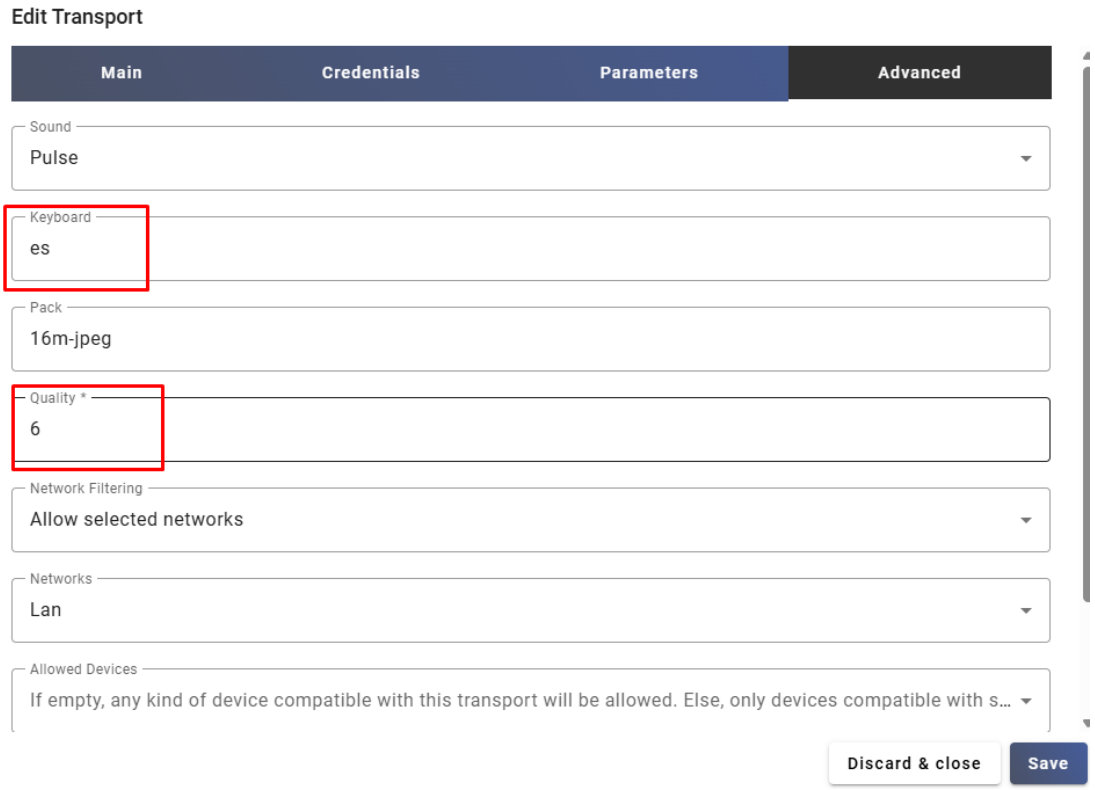


The Credentials tab will indicate the user with whom the application will run. In this case the uds user will be used (this user must exist in the Ubuntu template).



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

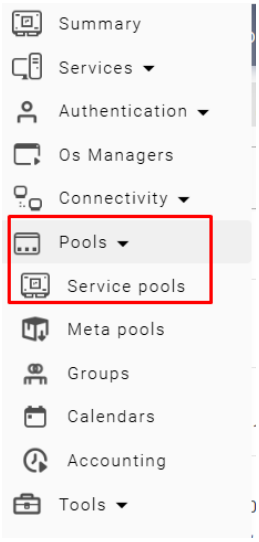
In the "advanced" tab, you enter the keyboard language and in "quality" you can define the image quality (0 - 9), being by default at 6. In addition, transport can be filtered by the networks that are defined and by type of device.



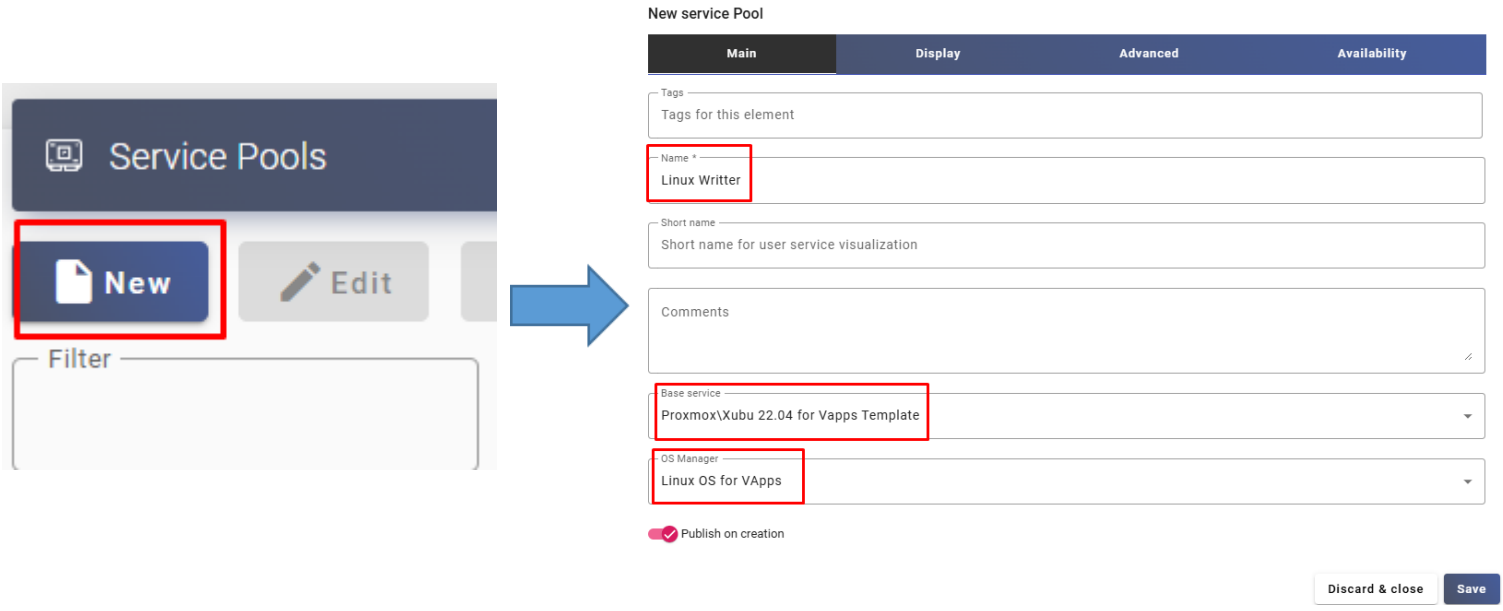
Once this configuration is done, "save".

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

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

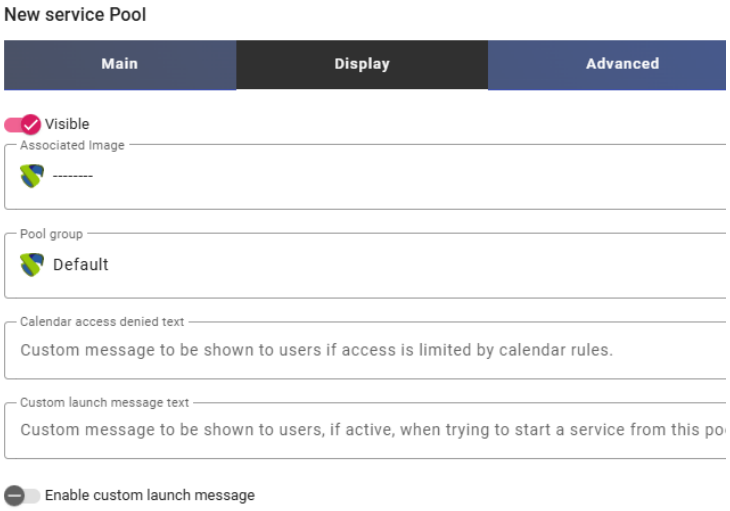


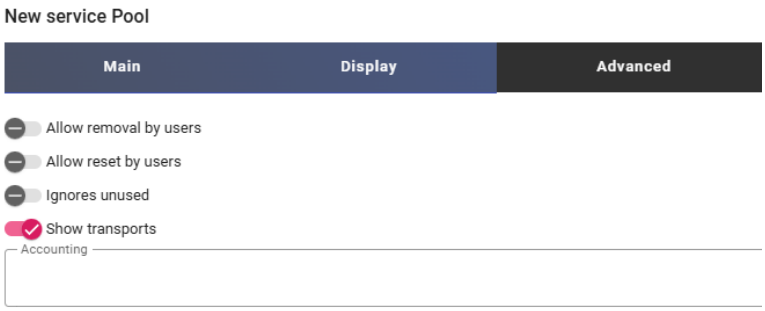
"New" is selected:



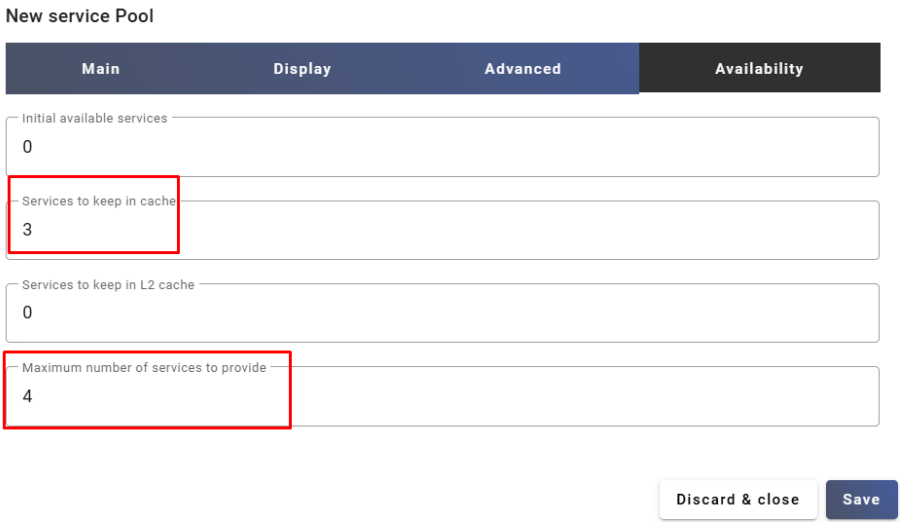
The name of the service will be indicated (this name will be visible to the user and will help them identify the service), the base service and the OS Manager (previously created) are selected.

In the "screen" tab you will indicate 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 you will leave the default checks.

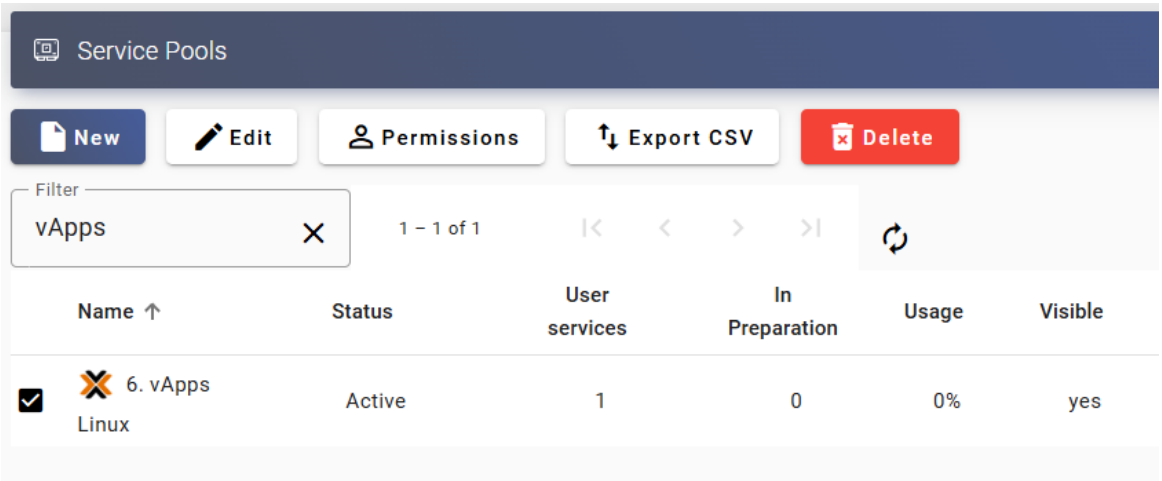




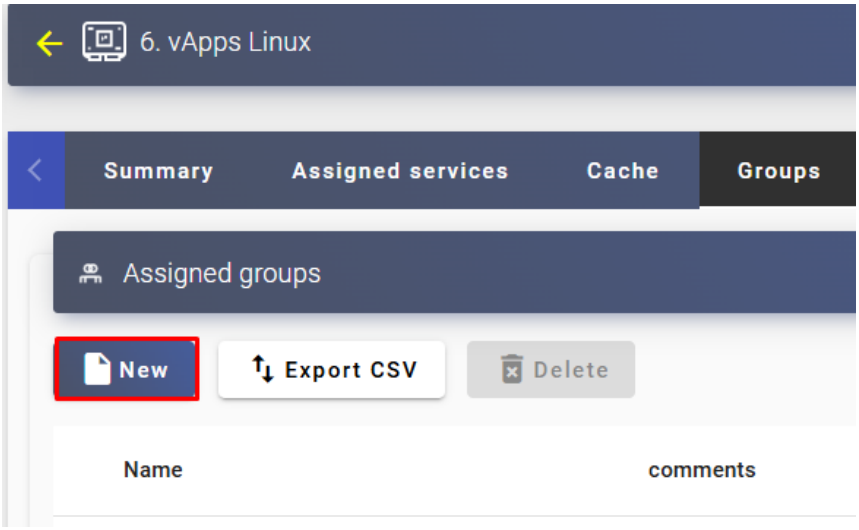
In the "availability" tab, you will have to indicate the number of services you want to have available for the connection of users (it is advisable to have as many services, initial or cached, as users are going to access the applications). Each user who connects will need a desktop and all the applications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



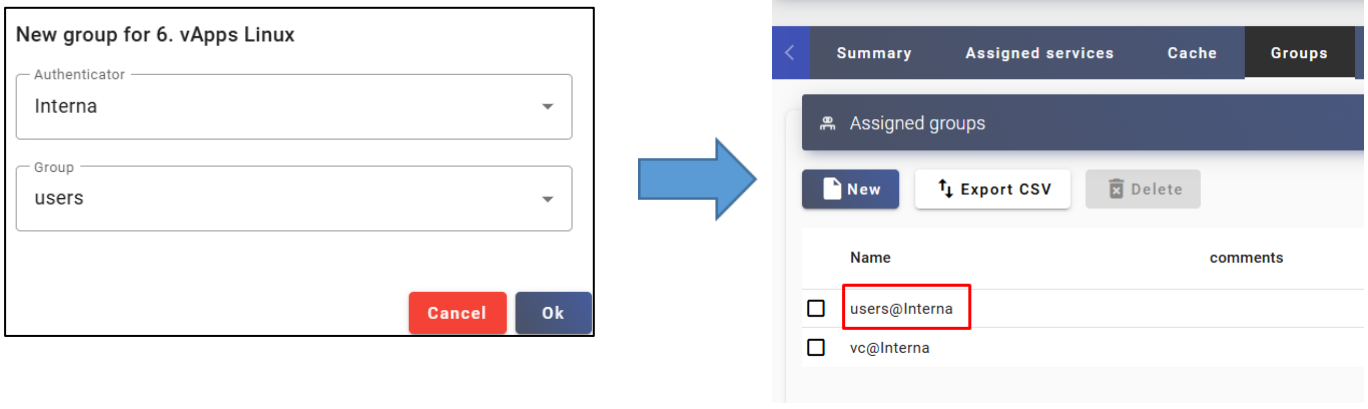
Once this configuration is done, it is saved to begin the creation of the service publication and the creation of the virtual desktops in which the virtual applications will be delivered.



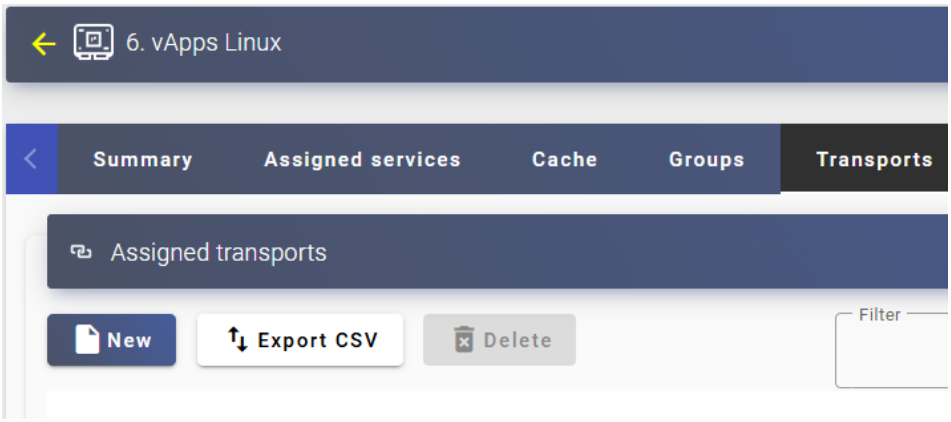
The next task will be to indicate which groups of users will access the virtual application service. Within the service pool created, you will access the "groups" tab.



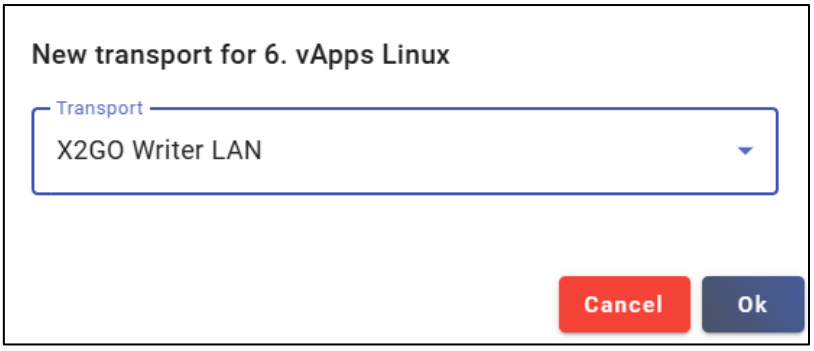
"New" is chosen and the authenticator/group (previously created) is selected and saved.

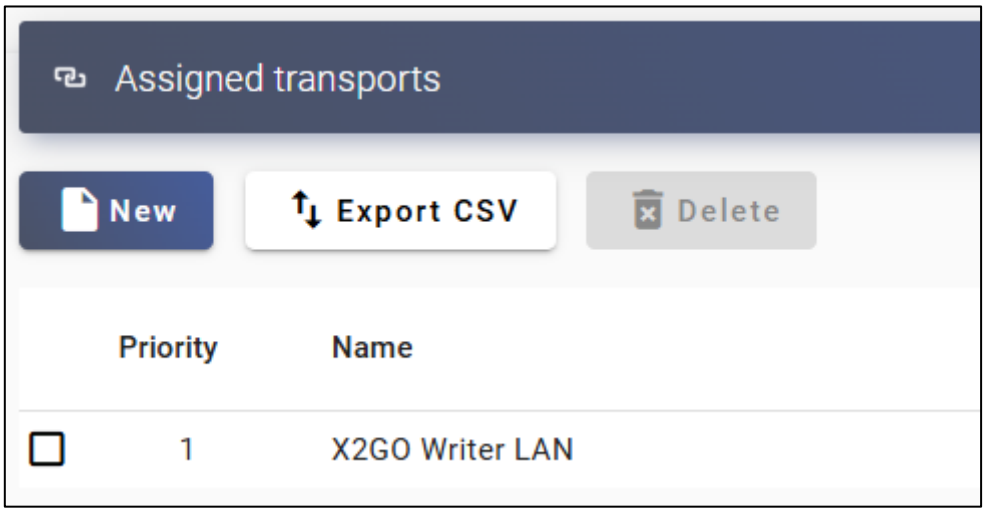


Once the group has been added, click on the "Transports" tab and assign the X2Go transport (where the route of the application to be virtualized has been defined) that has been previously created by clicking on "new".

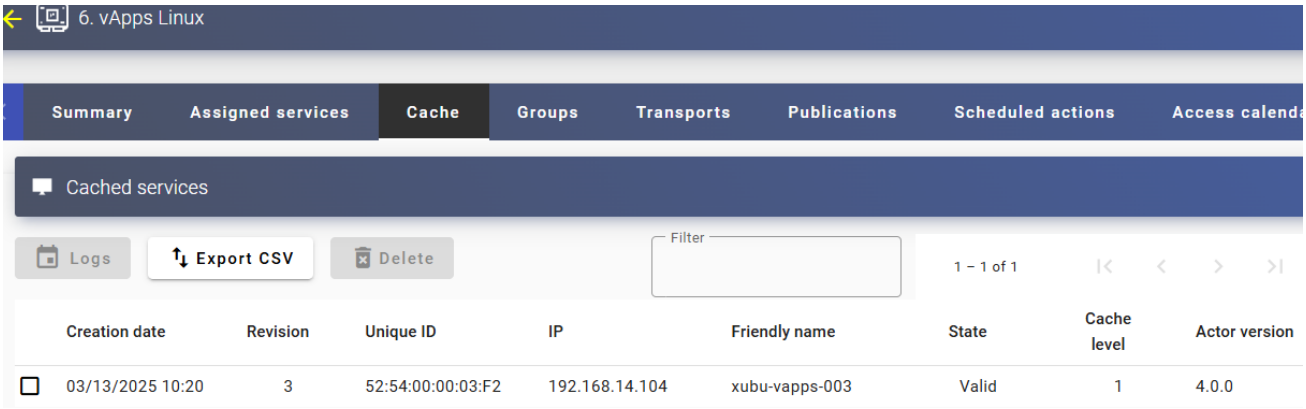


Select the transport and click on "save". A single transport or more than one may be added per service pool.



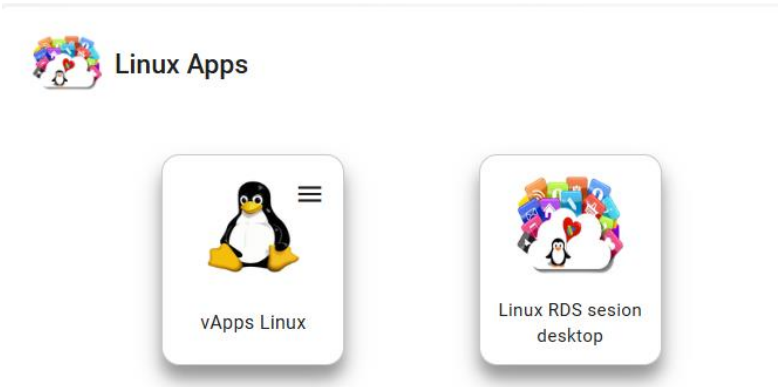


Before connecting to an application, it must be verified that the desktops autogenerated by UDS have been configured correctly. To do this, it will be checked in the "Cache" tab that we have the services in a valid state.



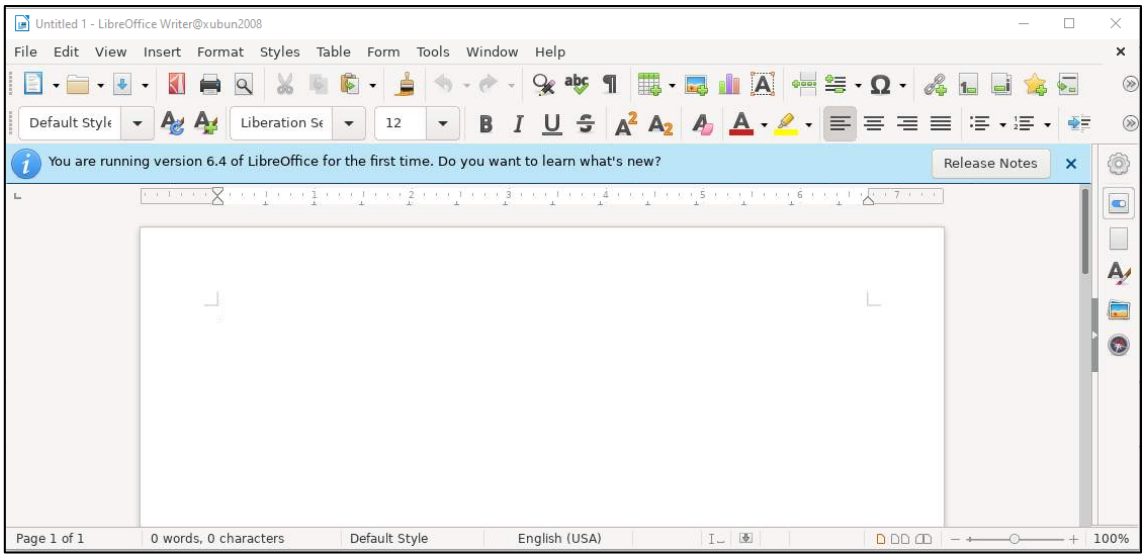
Access to Linux application services

Once all these configurations have been made, verified that the Linux desktops have been successfully auto generated and are in a valid state, users will be able to access the virtual application service.

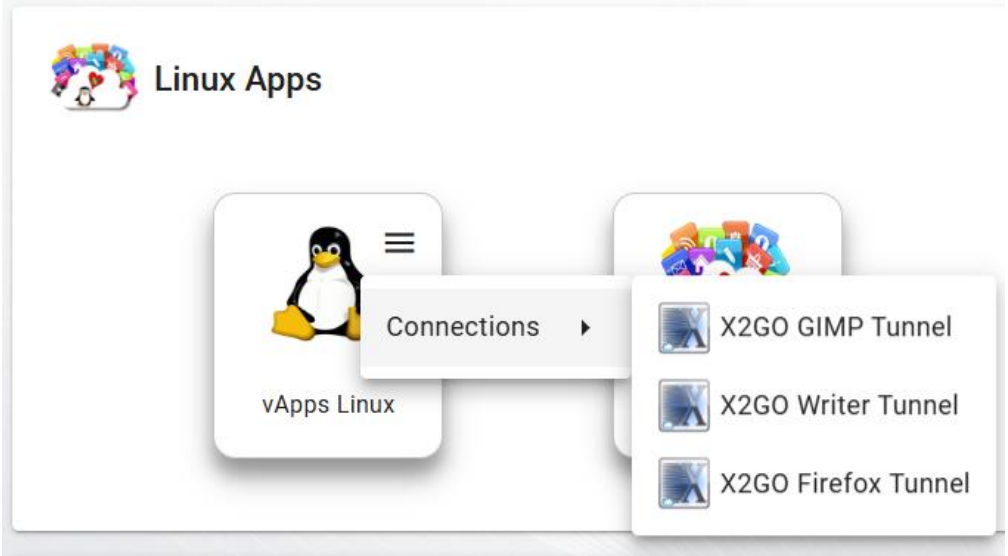


When the user logs in (they must have the X2Go client installed and the UDS client, for both Windows and Linux platforms), the following icon will appear on the taskbar:

And the app will open.



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



VIRTUAL CABLE'S SMART DIGITAL WORKPLACE SOLUTION

About UDS Enterprise

UDS Enterprise is a new software concept to create a **fully** customized **workplace virtualization** platform. It provides **24x7 secure access**, from anywhere and on any **device** to all the applications and software of an organization or educational center.

It allows virtualization **of Windows** and **Linux** desktops **and** applications **in a single console**, as well as **remote access** to Windows, Linux and macOS computers. Its Open Source base guarantees **compatibility with any** third-party technology. It can be deployed **on-premise**, in public, private, hybrid or **multicloud** cloud. Even **combine** multiple environments at the same time and perform **automatic, intelligent overflows** to optimize performance and efficiency. All with a **single subscription**.

About Virtual Cable

Virtual Cable is a company specialized in the **digital transformation** of the **workplace**. The company develops, supports and markets UDS Enterprise. It has recently been recognized as an ***IDC Innovator in Virtual Client Computing*** worldwide Its team of experts has designed **smart digital workplace solutions (VDI, vApp and remote access to physical computers)** tailored to **each sector** to provide a unique user experience fully adapted to the needs of each user profile. Virtual Cable's professionals have **more than 30 years of experience** in IT and software development and more than 15 years in virtualization technologies. Every day, **millions of Windows and Linux virtual desktops are deployed with UDS Enterprise around the world**.