CHALLENGE 6 HOTH

Julen Fernandez
Ibon Pina
SI2

Julen Fernandez
Ibon Pina
SI2

1-Task scheduler

1.1-Introduction

We are near to finish our work, but we still have some details in order to assure a more efficient work. We want to implement an HA system and use **VMWare** level **1 hypervisors,** in order to host our web server and a new machine that will perform some **repetitive tasks** such as making security copies or erasing some files. All these tasks will be performed using a periodically programmed series of scripts, executed from that virtual machines against our xxx.ally network.

Scripts are ideal for repetitive tasks and can be automatically executed with **crontab** in Linux, that's why we are gonna implement in our system a series of tasks that will create a backup every 30 minutes and will erase the temporary files of the web server machine.

This machine is going to be uploaded to the hypervisor altogether with the HTTP machine and its clone. For this purpose, **VMWare** is necessary for the correct conversion of the machine files.

1.2-Configuration

For the task scheduler, we are going to make two scripts, one of them is going to delete temporary unnecessary files from our HTTP machine, and the other one is going to copy the files of the /home folder from the HTTP machine every 30 minutes, excluding Yoda and then compress it.

These scripts are going to be running in the Task Scheduler machine, inside our class server that is running a level 1 hypervisor called ESXi. The IP of this machine is going to be 172.20.202.196. And after changing the IP we will execute the command
“service networking restart” in order to apply the new network configuration.

1.2.1-Temporal Files Script

In order to erase files from another machine, we will need to connect via ssh, but this command always asks us for the password, to fix this, we are going to use sshpass command, to install we will do “apt-get install sshpass”. To use this command we will write “sshpass -p “password”. So adding this line in front of every command it will automatically connect to the machine and execute the command.

temporal.sh

```
#Carpeta /tmp
sshpass -p 'M@ythe4th' ssh root@172.20.202.199 'rm -rf /tmp'
sshpass -p 'M@ythe4th' ssh root@172.20.202.199 'mkdir /tmp'
#Carpeta var/tmp
sshpass -p 'M@ythe4th' ssh root@172.20.202.199 'rm -rf /var/tmp'
sshpass -p 'M@ythe4th' ssh root@172.20.202.199 'mkdir /var/tmp'
```

1.2.2-Home Folder Backup Script

This script’s aim is to backup the folders inside the /home directory in our HTTP machine, without Yoda’s folder and compress it and save it with the correct name.

We will use the sshpass command again, as we have to do multiple connections via ssh to another machine.

First, we will create a temporary folder called backup and a folder called backups, then we copy all the files inside the /home folder of the HTTP machine to our /backup folder. Then we delete the folders that belong to yoda and to the local user “si2”.
The next step is to compress them into a tar.gz file using tar command.
In order to change the name to a more relevant one, we will define a variable called hora, which is going to store the date in a format showing the exact year - month - day -- hour - minute - second. Then we rename the file using mv and the variable.
Then we erase the /backup folder as we don’t need it anymore and we copy all the whole /backups folder where we have all our compressed files to the Task Scheduler machine using scp.

home.sh

```
#!/bin/bash
sshpass -p 'M@ythe4th' ssh root@172.20.202.199 'mkdir /backup'
sshpass -p 'M@ythe4th' ssh root@172.20.202.199 'mkdir /backups'
sshpass -p 'M@ythe4th' ssh root@172.20.202.199 'cp -r /home/* /backup'
sshpass -p 'M@ythe4th' ssh root@172.20.202.199 'cd /backup && rm -r yoda'
sshpass -p 'M@ythe4th' ssh root@172.20.202.199 'cd /backup && rm -r si2'
sshpass -p 'M@ythe4th' ssh root@172.20.202.199 'cd / && tar -czvf backuphome.tar.gz /backup'

sshpass -p 'M@ythe4th' ssh root@172.20.202.199 'cd / && hora=$(date +%F--%H-%M-%S) && mv backuphome.tar.gz /backups/backups/backuphome-$hora.tar.gz'

sshpass -p 'M@ythe4th' ssh root@172.20.202.199 'rm -r /backup'
sshpass -p 'M@ythe4th' scp -r root@172.20.202.199:/backups /backupshome
```

1.2.3-Script for executing scripts

We are going to make a third script that executes the two scripts that we have just created, this script is going to be very simple.

execute.sh

```
./home.sh
./temporal.sh
```

The final step is to add this script to crontab, where we will execute it every 30 minutes:
In order to edit the crontab file we will use this command:

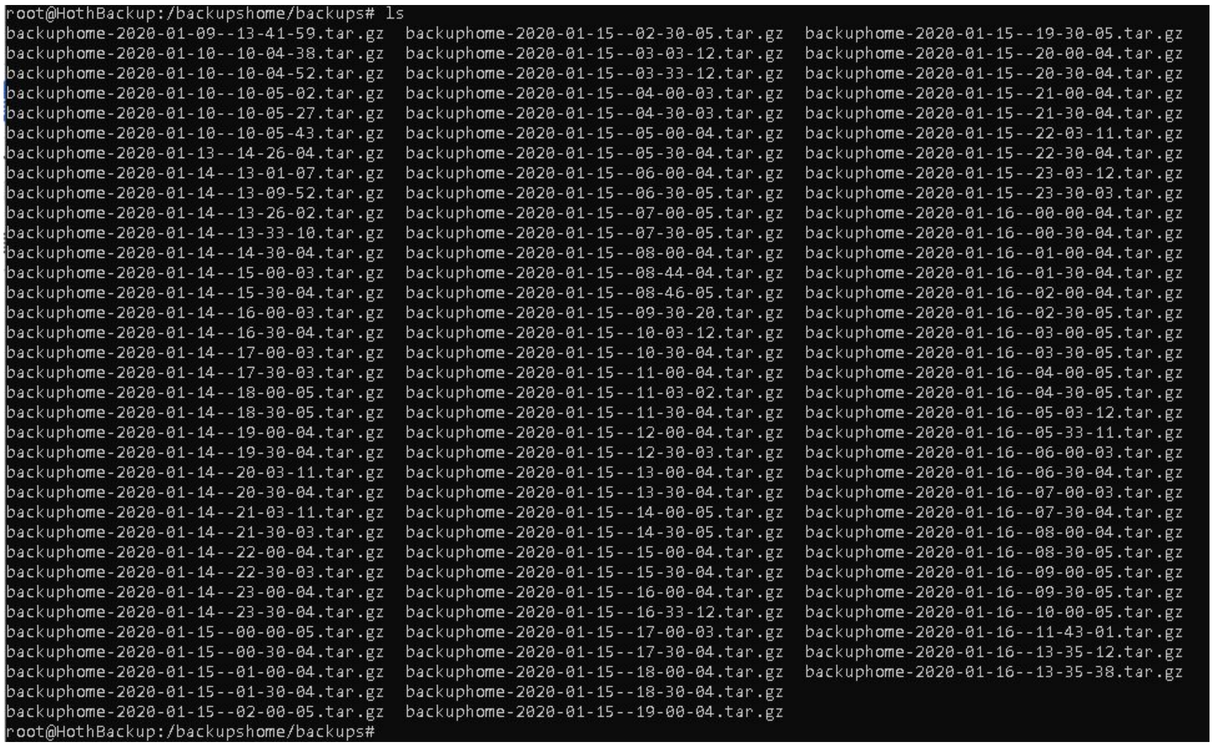
nano /etc/crontab

And we will add this line:

crontab

```
0/30 *  * * *   root    cd /root/scripts && ./execute.sh
```

With this line we will go to /root/scripts and execute the script “execute.sh” every 30 minutes of every day, every month and all the days of the week.



```
root@HothBackup:/backupshome/backups# ls
backuphome-2020-01-09--13-41-59.tar.gz  backuphome-2020-01-15--02-30-05.tar.gz  backuphome-2020-01-15--19-30-05.tar.gz
backuphome-2020-01-10--10-04-38.tar.gz  backuphome-2020-01-15--03-03-12.tar.gz  backuphome-2020-01-15--20-00-04.tar.gz
backuphome-2020-01-10--10-04-52.tar.gz  backuphome-2020-01-15--03-33-12.tar.gz  backuphome-2020-01-15--20-30-04.tar.gz
backuphome-2020-01-10--10-05-02.tar.gz  backuphome-2020-01-15--04-00-03.tar.gz  backuphome-2020-01-15--21-00-04.tar.gz
backuphome-2020-01-10--10-05-27.tar.gz  backuphome-2020-01-15--04-30-03.tar.gz  backuphome-2020-01-15--21-30-04.tar.gz
backuphome-2020-01-10--10-05-43.tar.gz  backuphome-2020-01-15--05-00-04.tar.gz  backuphome-2020-01-15--22-03-11.tar.gz
backuphome-2020-01-13--14-26-04.tar.gz  backuphome-2020-01-15--05-30-04.tar.gz  backuphome-2020-01-15--22-30-04.tar.gz
backuphome-2020-01-14--13-01-07.tar.gz  backuphome-2020-01-15--06-00-04.tar.gz  backuphome-2020-01-15--23-03-12.tar.gz
backuphome-2020-01-14--13-09-52.tar.gz  backuphome-2020-01-15--06-30-05.tar.gz  backuphome-2020-01-15--23-30-03.tar.gz
backuphome-2020-01-14--13-26-02.tar.gz  backuphome-2020-01-15--07-00-05.tar.gz  backuphome-2020-01-16--00-00-04.tar.gz
backuphome-2020-01-14--13-33-10.tar.gz  backuphome-2020-01-15--07-30-05.tar.gz  backuphome-2020-01-16--00-30-04.tar.gz
backuphome-2020-01-14--14-30-04.tar.gz  backuphome-2020-01-15--08-00-04.tar.gz  backuphome-2020-01-16--01-00-04.tar.gz
backuphome-2020-01-14--15-00-03.tar.gz  backuphome-2020-01-15--08-44-04.tar.gz  backuphome-2020-01-16--01-30-04.tar.gz
backuphome-2020-01-14--15-30-04.tar.gz  backuphome-2020-01-15--08-46-05.tar.gz  backuphome-2020-01-16--02-00-04.tar.gz
backuphome-2020-01-14--16-00-03.tar.gz  backuphome-2020-01-15--09-30-20.tar.gz  backuphome-2020-01-16--02-30-05.tar.gz
backuphome-2020-01-14--16-30-04.tar.gz  backuphome-2020-01-15--10-03-12.tar.gz  backuphome-2020-01-16--03-00-05.tar.gz
backuphome-2020-01-14--17-00-03.tar.gz  backuphome-2020-01-15--10-30-04.tar.gz  backuphome-2020-01-16--03-30-05.tar.gz
backuphome-2020-01-14--17-30-03.tar.gz  backuphome-2020-01-15--11-00-04.tar.gz  backuphome-2020-01-16--04-00-05.tar.gz
backuphome-2020-01-14--18-00-05.tar.gz  backuphome-2020-01-15--11-03-02.tar.gz  backuphome-2020-01-16--04-30-05.tar.gz
backuphome-2020-01-14--18-30-05.tar.gz  backuphome-2020-01-15--11-30-04.tar.gz  backuphome-2020-01-16--05-03-12.tar.gz
backuphome-2020-01-14--19-00-04.tar.gz  backuphome-2020-01-15--12-00-04.tar.gz  backuphome-2020-01-16--05-33-11.tar.gz
backuphome-2020-01-14--19-30-04.tar.gz  backuphome-2020-01-15--12-30-03.tar.gz  backuphome-2020-01-16--06-00-03.tar.gz
backuphome-2020-01-14--20-03-11.tar.gz  backuphome-2020-01-15--13-00-04.tar.gz  backuphome-2020-01-16--06-30-04.tar.gz
backuphome-2020-01-14--20-30-04.tar.gz  backuphome-2020-01-15--13-30-04.tar.gz  backuphome-2020-01-16--07-00-03.tar.gz
backuphome-2020-01-14--21-03-11.tar.gz  backuphome-2020-01-15--14-00-05.tar.gz  backuphome-2020-01-16--07-30-04.tar.gz
backuphome-2020-01-14--21-30-03.tar.gz  backuphome-2020-01-15--14-30-05.tar.gz  backuphome-2020-01-16--08-00-04.tar.gz
backuphome-2020-01-14--22-00-04.tar.gz  backuphome-2020-01-15--15-00-04.tar.gz  backuphome-2020-01-16--08-30-05.tar.gz
backuphome-2020-01-14--22-30-03.tar.gz  backuphome-2020-01-15--15-30-04.tar.gz  backuphome-2020-01-16--09-00-05.tar.gz
backuphome-2020-01-14--23-00-04.tar.gz  backuphome-2020-01-15--16-00-04.tar.gz  backuphome-2020-01-16--09-30-05.tar.gz
backuphome-2020-01-14--23-30-04.tar.gz  backuphome-2020-01-15--16-33-12.tar.gz  backuphome-2020-01-16--10-00-05.tar.gz
backuphome-2020-01-15--00-00-05.tar.gz  backuphome-2020-01-15--17-00-03.tar.gz  backuphome-2020-01-16--11-43-01.tar.gz
backuphome-2020-01-15--00-30-04.tar.gz  backuphome-2020-01-15--17-30-04.tar.gz  backuphome-2020-01-16--13-35-12.tar.gz
backuphome-2020-01-15--01-00-04.tar.gz  backuphome-2020-01-15--18-00-04.tar.gz  backuphome-2020-01-16--13-35-38.tar.gz
backuphome-2020-01-15--01-30-04.tar.gz  backuphome-2020-01-15--18-30-04.tar.gz
backuphome-2020-01-15--02-00-05.tar.gz  backuphome-2020-01-15--19-00-04.tar.gz
root@HothBackup:/backupshome/backups#
```

2-High availability

2.1-Introduction

There is an extra difficulty: that web server is very **critical for our systems**, because Siths are searching different exploits and our server is settled by their attacks. Thus, we are doing an extra economical and configuration effort setting our system in HA, so it can support the fall of one of the servers and we continue having the service active, taking advantage of clustering **(Pacemaker and Corosync).**

With this characteristic, we can ensure that the performance level of the system is going to improve as it is going to be available and accessible more time. In case our main machine (With each correspondent services) is attacked or is not accessible when an error happens, with the high availability activated the system won't go down and the services will remain active. The user won't notice the failure but the maintenance activity will.

Pacemaker and Corosync are the services we are going to use.

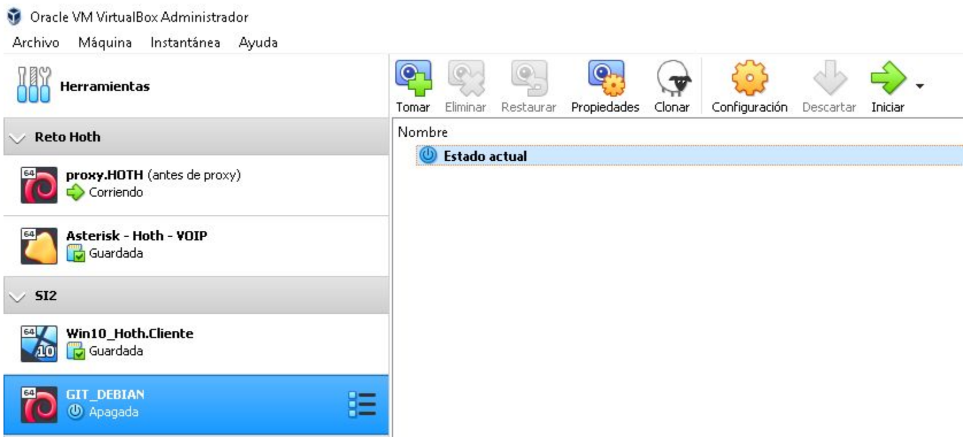
2.2-Configuration

For the high availability part we are going to use two machines, the first one is going to be the HTTP/FTP machine, and the second one is going to be a clone of the first one, these two machines are going to be located in the class server running ESXi. In order to upload these two machines, along with the Task Scheduler machine we will have to take some steps as you can't just export them directly to ESXi.

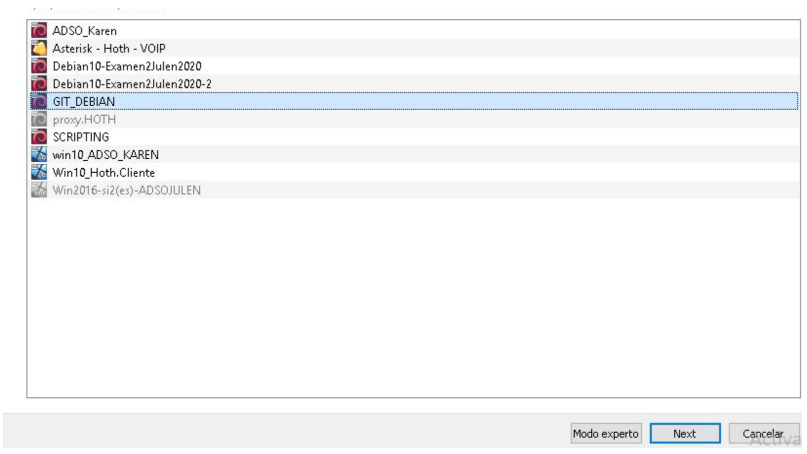
These two machines ip's are 172.20.202.199 for the main one, and 172.20.202.195 for the clone of the main machine. Both are going to be running an apache server, and the aim of this part of the challenge is to have always an apache machine running, even if the main one fails.

2.2.1-Exporting VM's from VirtualBox to ESXi

The first step is to go to VirtualBox and click on the machine we want to export, then we click on "Archivo" and "exportar servicio virtualizado"

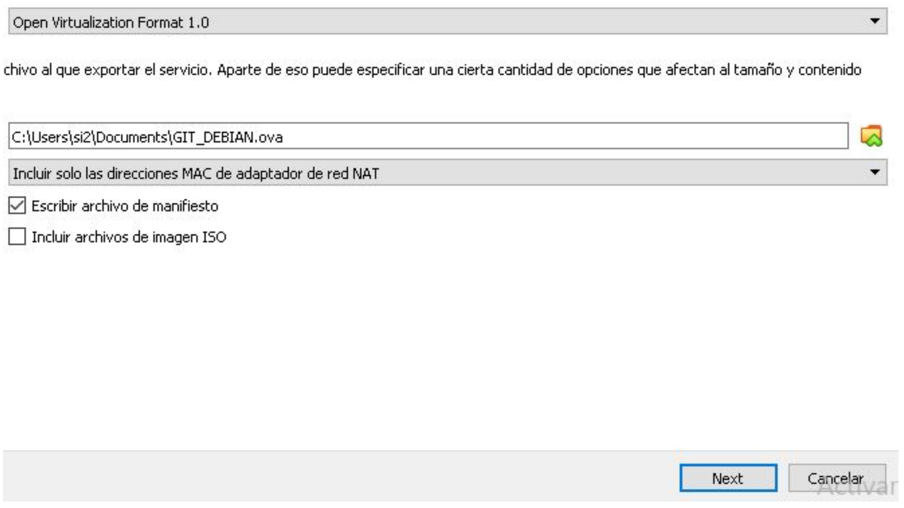


We click on "next"



Julen Fernandez
Ibon Pina
SI2

We let the open virtualization format to version 1.0 and click on next



Open Virtualization Format 1.0

chivo al que exportar el servicio. Aparte de eso puede especificar una cierta cantidad de opciones que afectan al tamaño y contenido

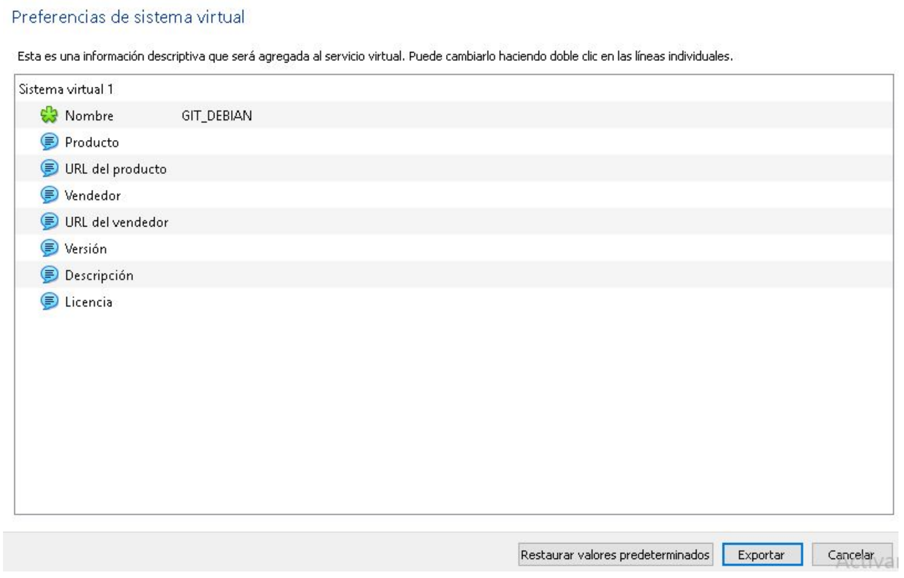
C:\Users\si2\Documents\GIT_DEBIAN.ova

Incluir solo las direcciones MAC de adaptador de red NAT

☑ Escribir archivo de manifiesto

☐ Incluir archivos de imagen ISO

Finally, we click on “exportar” to export the machine into an ova file.



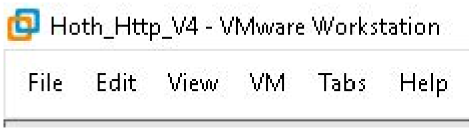
Preferencias de sistema virtual

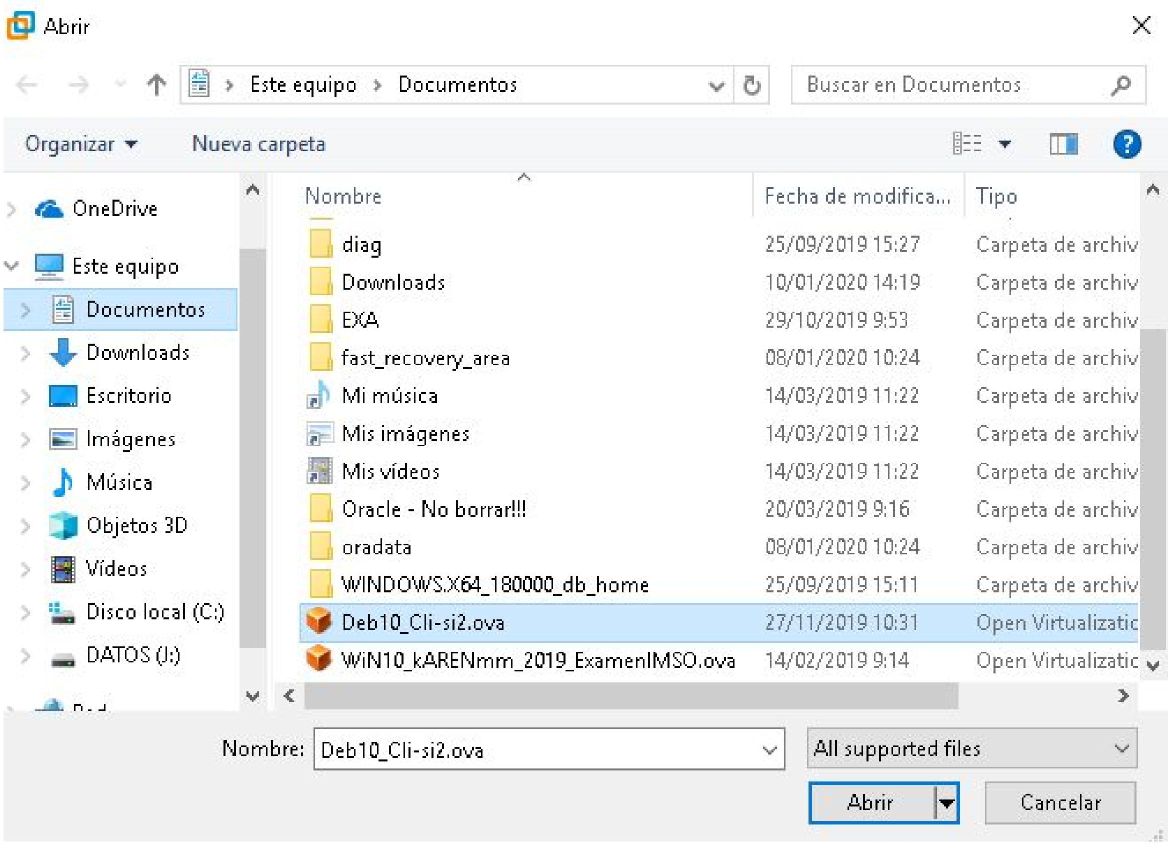
Esta es una información descriptiva que será agregada al servicio virtual. Puede cambiarlo haciendo doble clic en las líneas individuales.

Sistema virtual 1

Nombre	GIT_DEBIAN
Producto	
URL del producto	
Vendedor	
URL del vendedor	
Versión	
Descripción	
Licencia	

The next step is to open VMware Workstation, and click on “File”, then “open” and we select the ova: (This is just an example ova and exportation)





Abrir

Este equipo > Documentos

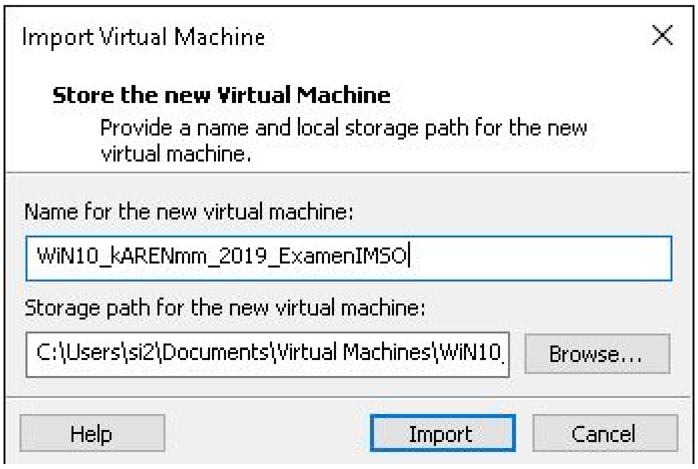
Buscar en Documentos

Organizar ▾ Nueva carpeta

Nombre	Fecha de modifica...	Tipo
diag	25/09/2019 15:27	Carpeta de archiv
Downloads	10/01/2020 14:19	Carpeta de archiv
EXA	29/10/2019 9:53	Carpeta de archiv
fast_recovery_area	08/01/2020 10:24	Carpeta de archiv
Mi música	14/03/2019 11:22	Carpeta de archiv
Mis imágenes	14/03/2019 11:22	Carpeta de archiv
Mis vídeos	14/03/2019 11:22	Carpeta de archiv
Oracle - No borrar!!!	20/03/2019 9:16	Carpeta de archiv
oradata	08/01/2020 10:24	Carpeta de archiv
WINDOWS.X64_180000_db_home	25/09/2019 15:11	Carpeta de archiv
Deb10_Cli-si2.ova	27/11/2019 10:31	Open Virtualizatic
WiN10_kARENmm_2019_ExamenIMSO.ova	14/02/2019 9:14	Open Virtualizatic

Nombre: Deb10_Cli-si2.ova

All supported files



Import Virtual Machine

Store the new Virtual Machine

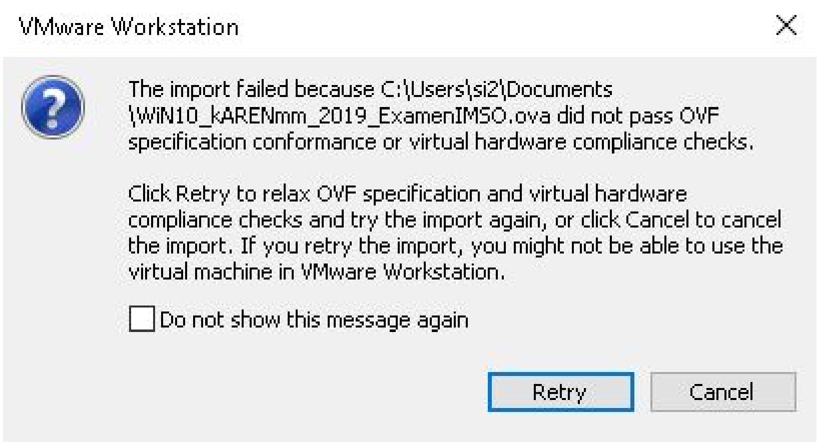
Provide a name and local storage path for the new virtual machine.

Name for the new virtual machine: WiN10_kARENmm_2019_ExamenIMSO

Storage path for the new virtual machine: C:\Users\si2\Documents\Virtual Machines\WiN10

It will give us an error, we just click on “Retry”

Julen Fernandez
Ibon Pina
SI2

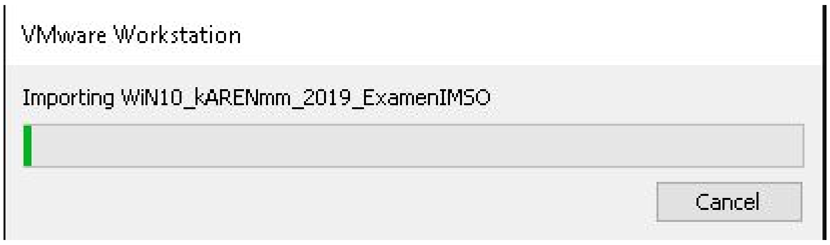

VMware Workstation

The import failed because C:\Users\si2\Documents \WiN10_kARENmm_2019_ExamenIMSO.ova did not pass OVF specification conformance or virtual hardware compliance checks.

Click Retry to relax OVF specification and virtual hardware compliance checks and try the import again, or click Cancel to cancel the import. If you retry the import, you might not be able to use the virtual machine in VMware Workstation.

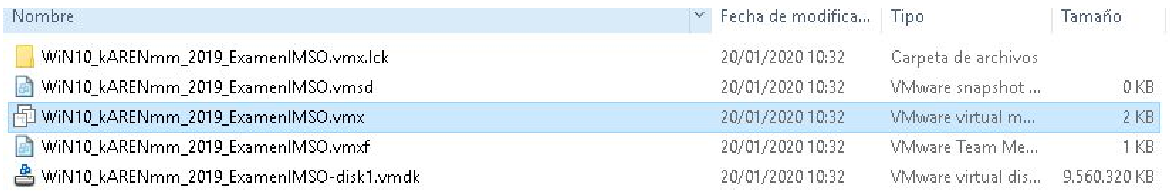
☐ Do not show this message again

It starts importing


VMware Workstation

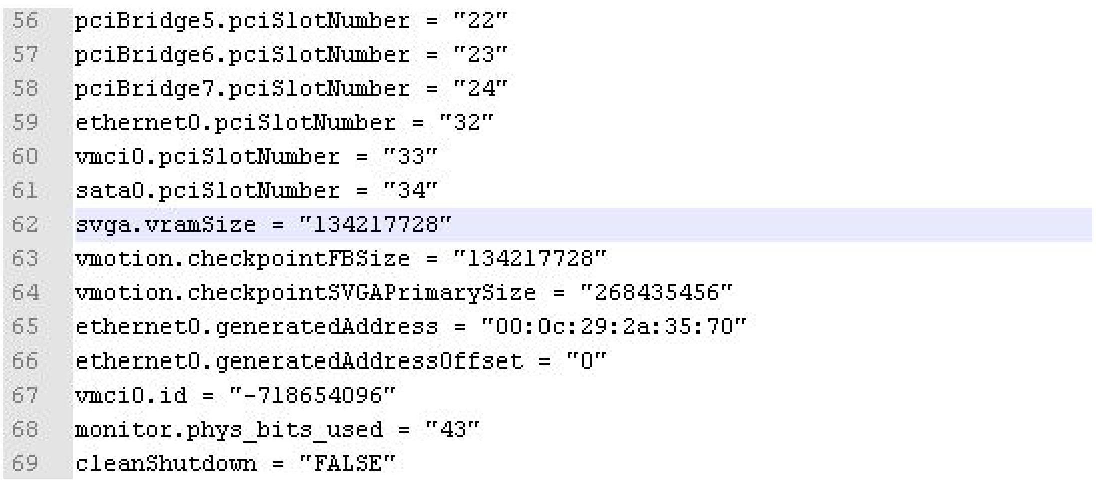
Importing WiN10_kARENmm_2019_ExamenIMSO

Once is added to VMWare, we will turn it on, and then turn it off. We have to do this because the line that we have to edit later in order to make the machine work on ESXi won’t appear unless we turn it on at least once inside VMWare workstation. After turning it off we will go to the machine’s location folder and edit the .vmx file with any text editor.



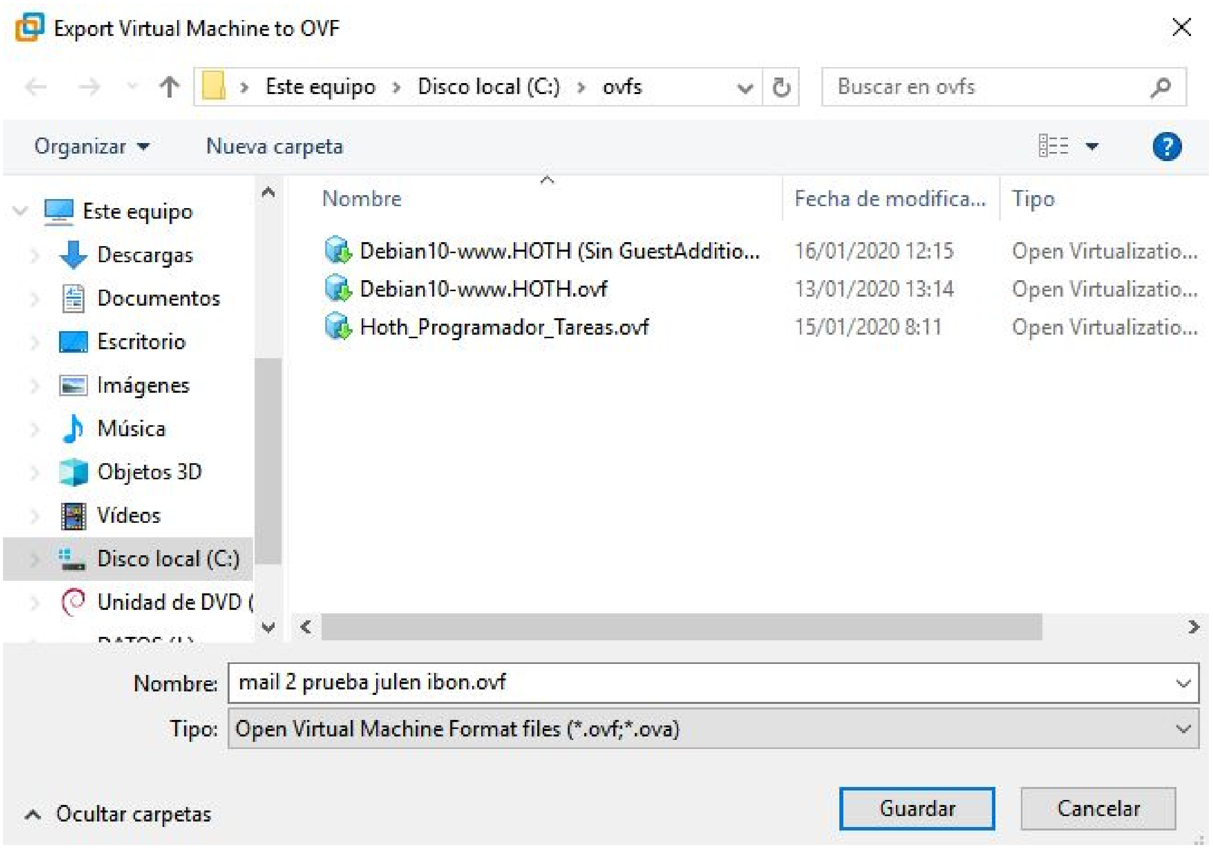
Nombre	Fecha de modifica...	Tipo	Tamaño
WiN10_kARENmm_2019_ExamenIMSO.vmx.lck	20/01/2020 10:32	Carpeta de archivos	
WiN10_kARENmm_2019_ExamenIMSO.vmsd	20/01/2020 10:32	VMware snapshot ...	0 KB
WiN10_kARENmm_2019_ExamenIMSO.vmx	20/01/2020 10:32	VMware virtual m...	2 KB
WiN10_kARENmm_2019_ExamenIMSO.vmxf	20/01/2020 10:32	VMware Team Me...	1 KB
WiN10_kARENmm_2019_ExamenIMSO-disk1.vmdk	20/01/2020 10:32	VMware virtual dis...	9.560.320 KB

We just need to set the value of “svga.vramSize” to “134217728” instead of “268435456”



```
pciBridge5.pciSlotNumber = "22"
pciBridge6.pciSlotNumber = "23"
pciBridge7.pciSlotNumber = "24"
ethernet0.pciSlotNumber = "32"
vmci0.pciSlotNumber = "33"
sata0.pciSlotNumber = "34"
svga.vramSize = "134217728"
vmotion.checkpointFBSize = "134217728"
vmotion.checkpointSVGAPrimarySize = "268435456"
ethernet0.generatedAddress = "00:0c:29:2a:35:70"
ethernet0.generatedAddressOffset = "0"
vmci0.id = "-718654096"
monitor.phys_bits_used = "43"
cleanShutdown = "FALSE"
```

Finally, we export the machine from VMWare workstation by clicking on “file” and then “export to .ovf”, then we save it in any folder:

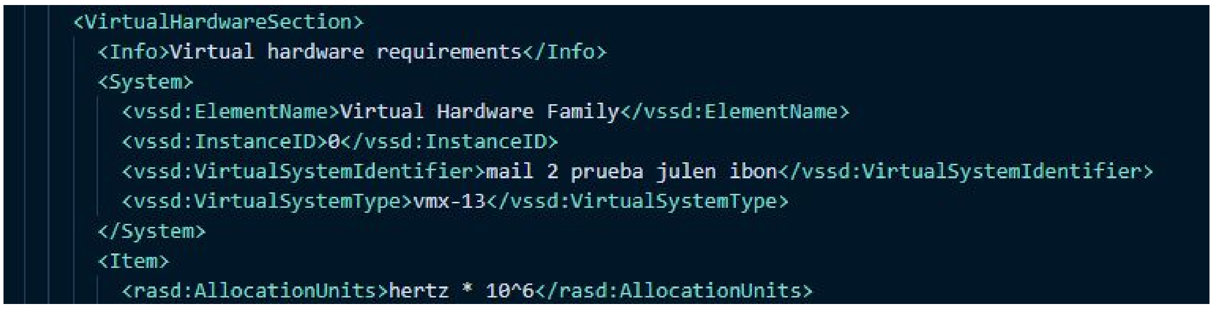


Nombre	Fecha de modifica...	Tipo
Debian10-www.HOTH (Sin GuestAdditio...	16/01/2020 12:15	Open Virtualizatio...
Debian10-www.HOTH.ovf	13/01/2020 13:14	Open Virtualizatio...
Hoth_Programador_Tareas.ovf	15/01/2020 8:11	Open Virtualizatio...

Nombre: mail 2 prueba julen ibon.ovf

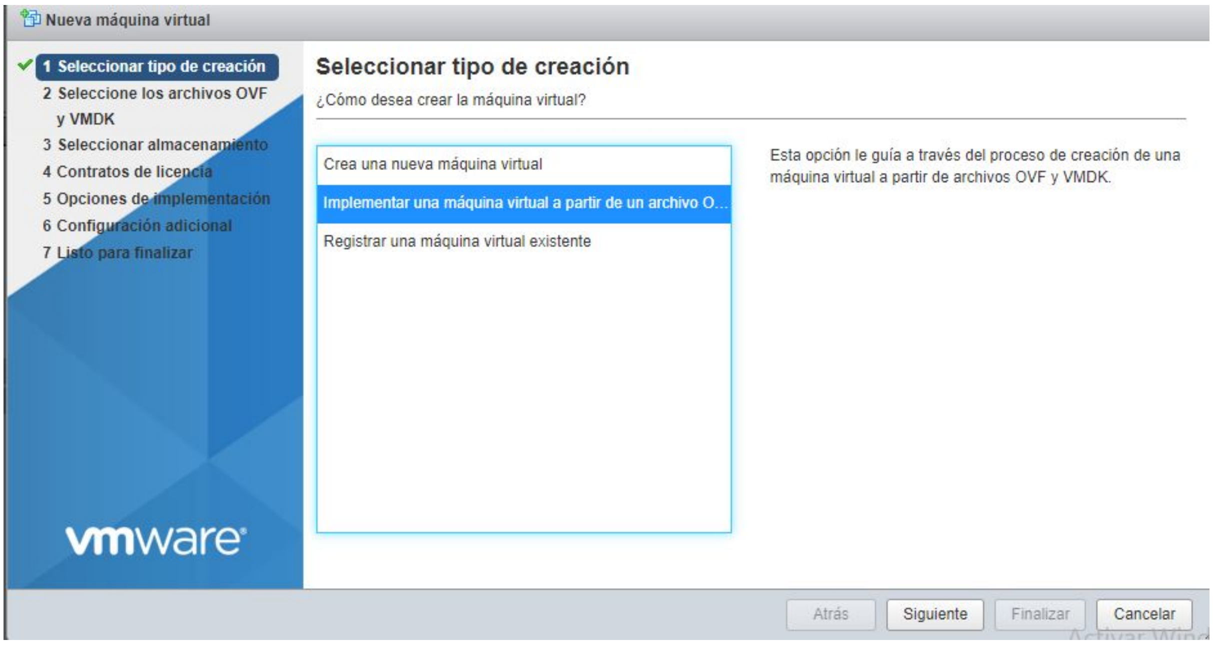
Tipo: Open Virtual Machine Format files (*.ovf;*.ova)

After exporting the machine we will get 3 files. For the upload of the machine to ESXi, we will only need two of them, the .ovf file and the .vmdk file.
Once we have our .ovf file, we have to edit it in order to work properly in ESXi, for this, we can open it with any text editor, and change the following:
We only have to change the virtualsystemtype from vmx-15 to vmx-13 and then save it.



```
<VirtualHardwareSection>
  <Info>Virtual hardware requirements</Info>
  <System>
    <vssd:ElementName>Virtual Hardware Family</vssd:ElementName>
    <vssd:InstanceID>0</vssd:InstanceID>
    <vssd:VirtualSystemIdentifier>mail 2 prueba julen ibon</vssd:VirtualSystemIdentifier>
    <vssd:VirtualSystemType>vmx-13</vssd:VirtualSystemType>
  </System>
  <Item>
    <rasd:AllocationUnits>hertz * 10^6</rasd:AllocationUnits>
```

In order to add a new machine inside ESXi the first step is to connect via web to the server, using the ip 172.20.202.70 and the username and password, then we have to go to the zone were all the virtual machines are, and we will click on “Crear/Registrar máquina virtual”.
We select the second option:



Nueva máquina virtual

Seleccionar tipo de creación

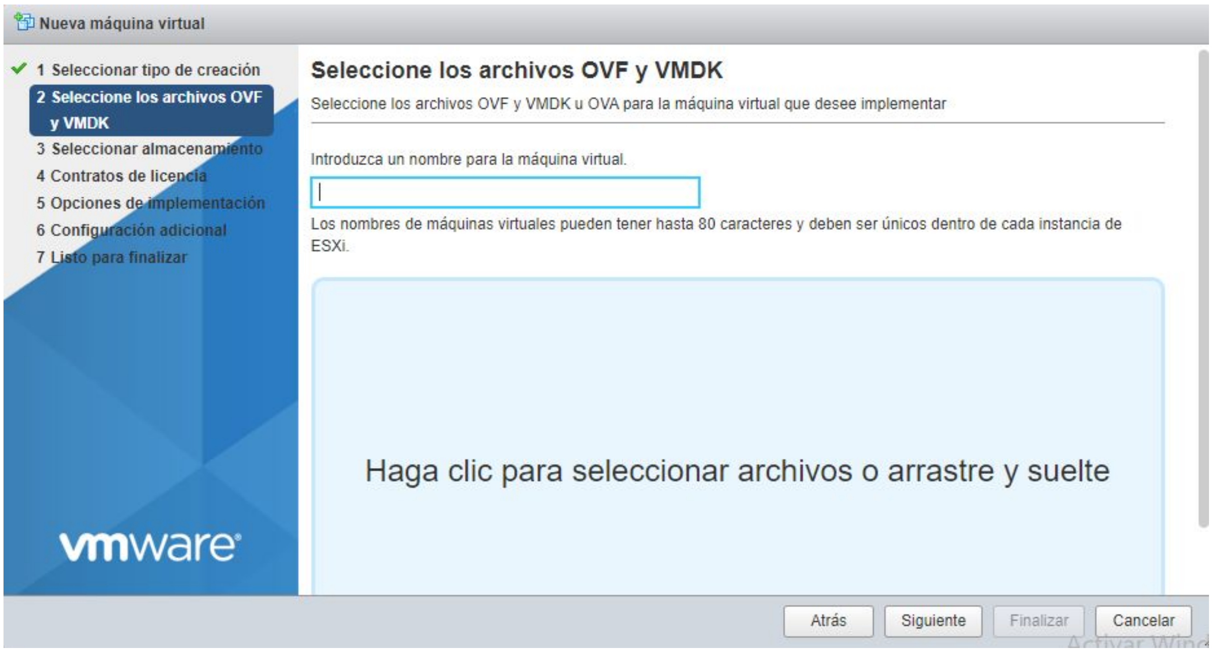
¿Cómo desea crear la máquina virtual?

- Crea una nueva máquina virtual
- Implementar una máquina virtual a partir de un archivo O...
- Registrar una máquina virtual existente

Esta opción le guía a través del proceso de creación de una máquina virtual a partir de archivos OVF y VMDK.

1 Seleccionar tipo de creación
2 Seleccione los archivos OVF y VMDK
3 Seleccionar almacenamiento
4 Contratos de licencia
5 Opciones de implementación
6 Configuración adicional
7 Listo para finalizar
vmware
Atrás
Siguiente
Finalizar
Cancelar

Now we can set the name for the new machine and select the files needed, the ovf and vmdk files.


Nueva máquina virtual
1 Seleccionar tipo de creación
2 Seleccione los archivos OVF y VMDK
3 Seleccionar almacenamiento
4 Contratos de licencia
5 Opciones de implementación
6 Configuración adicional
7 Listo para finalizar
Atrás
Siguiente
Finalizar
Cancelar

Seleccione los archivos OVF y VMDK

Seleccione los archivos OVF y VMDK u OVA para la máquina virtual que desee implementar

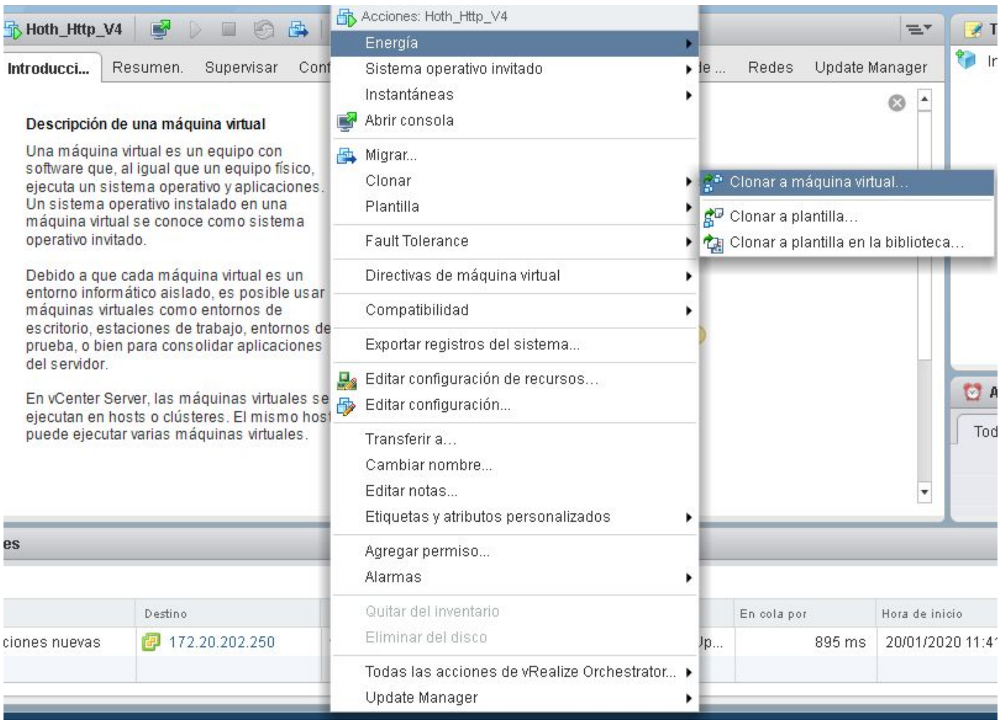
Introduzca un nombre para la máquina virtual.

Los nombres de máquinas virtuales pueden tener hasta 80 caracteres y deben ser únicos dentro de cada instancia de ESXi.

Haga clic para seleccionar archivos o arrastre y suelte

We click on next and finally on “Finalizar”, the machine will take a few minutes to upload.
Once we have the main HTTP machine uploaded we will make a clone out of it:

If we want to make a clone, we have to connect to vsphere, where we can manage all the machines for the server and also clone them. The IP for vsphere is 172.20.202.250.
Once there we have to select the machine that we want to clone.
We click on “Acciones” and then on “Clonar”



Descripción de una máquina virtual

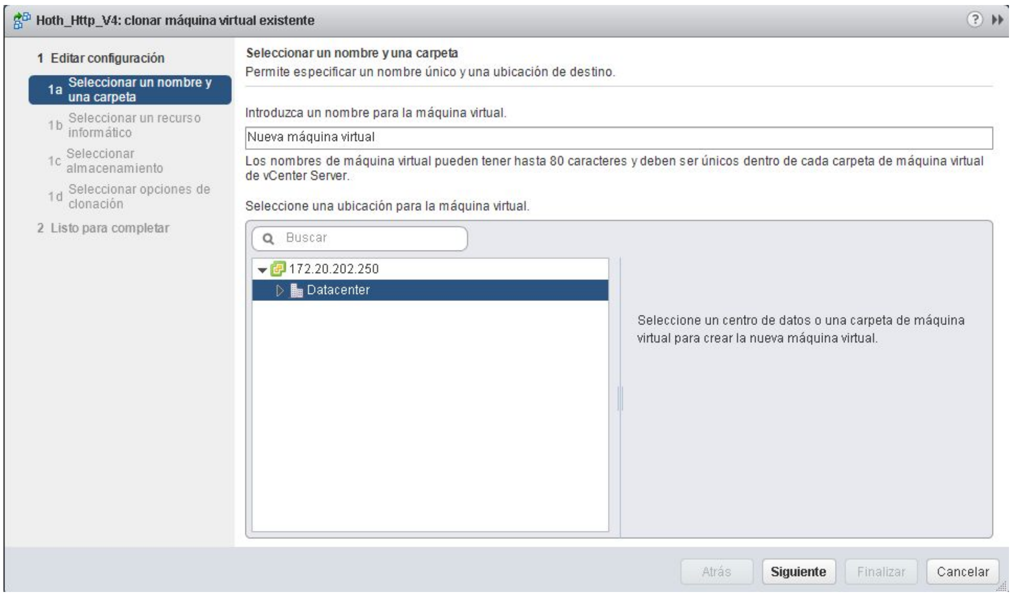
Una máquina virtual es un equipo con software que, al igual que un equipo físico, ejecuta un sistema operativo y aplicaciones. Un sistema operativo instalado en una máquina virtual se conoce como sistema operativo invitado.

Debido a que cada máquina virtual es un entorno informático aislado, es posible usar máquinas virtuales como entornos de escritorio, estaciones de trabajo, entornos de prueba, o bien para consolidar aplicaciones del servidor.

En vCenter Server, las máquinas virtuales se ejecutan en hosts o clústeres. El mismo host puede ejecutar varias máquinas virtuales.

Julen Fernandez
Ibon Pina
SI2

This window pops up, and here we can type the name of the cloned machine and where it is going to be stored:



Hoth_Http_V4: clonar máquina virtual existente

1 Editar configuración
1a Seleccionar un nombre y una carpeta
1b Seleccionar un recurso informático
1c Seleccionar almacenamiento
1d Seleccionar opciones de clonación
2 Listo para completar

Seleccionar un nombre y una carpeta
Permite especificar un nombre único y una ubicación de destino.

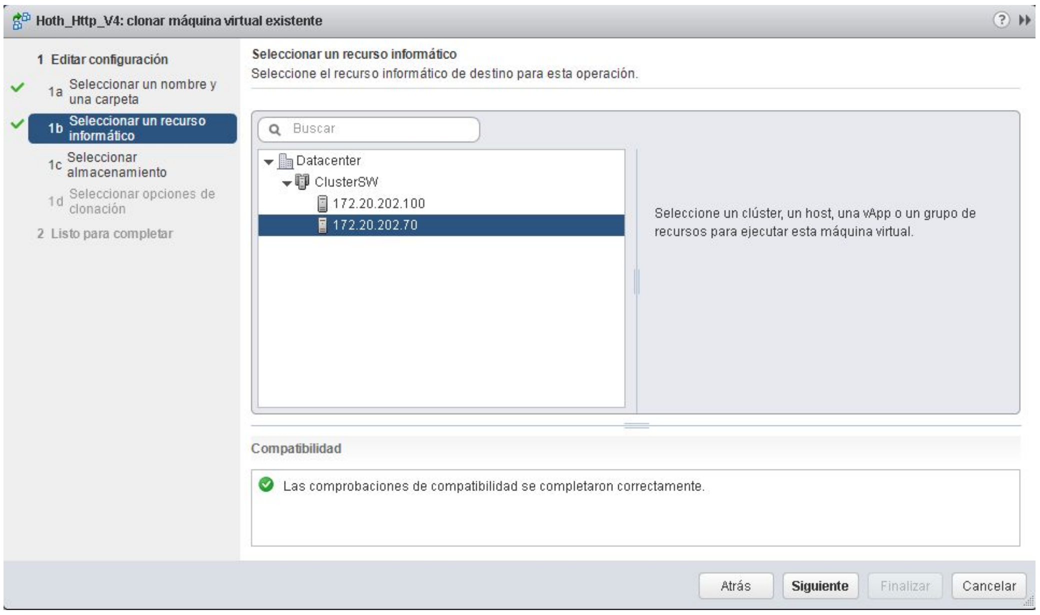
Introduzca un nombre para la máquina virtual.
Nueva máquina virtual
Los nombres de máquina virtual pueden tener hasta 80 caracteres y deben ser únicos dentro de cada carpeta de máquina virtual de vCenter Server.

Seleccione una ubicación para la máquina virtual.
Buscar
172.20.202.250
Datacenter

Seleccione un centro de datos o una carpeta de máquina virtual para crear la nueva máquina virtual.

Atrás | Siguiente | Finalizar | Cancelar

We now have to select the server where we want to clone the new machine



Hoth_Http_V4: clonar máquina virtual existente

1 Editar configuración
1a Seleccionar un nombre y una carpeta
1b Seleccionar un recurso informático
1c Seleccionar almacenamiento
1d Seleccionar opciones de clonación
2 Listo para completar

Seleccionar un recurso informático
Seleccione el recurso informático de destino para esta operación.

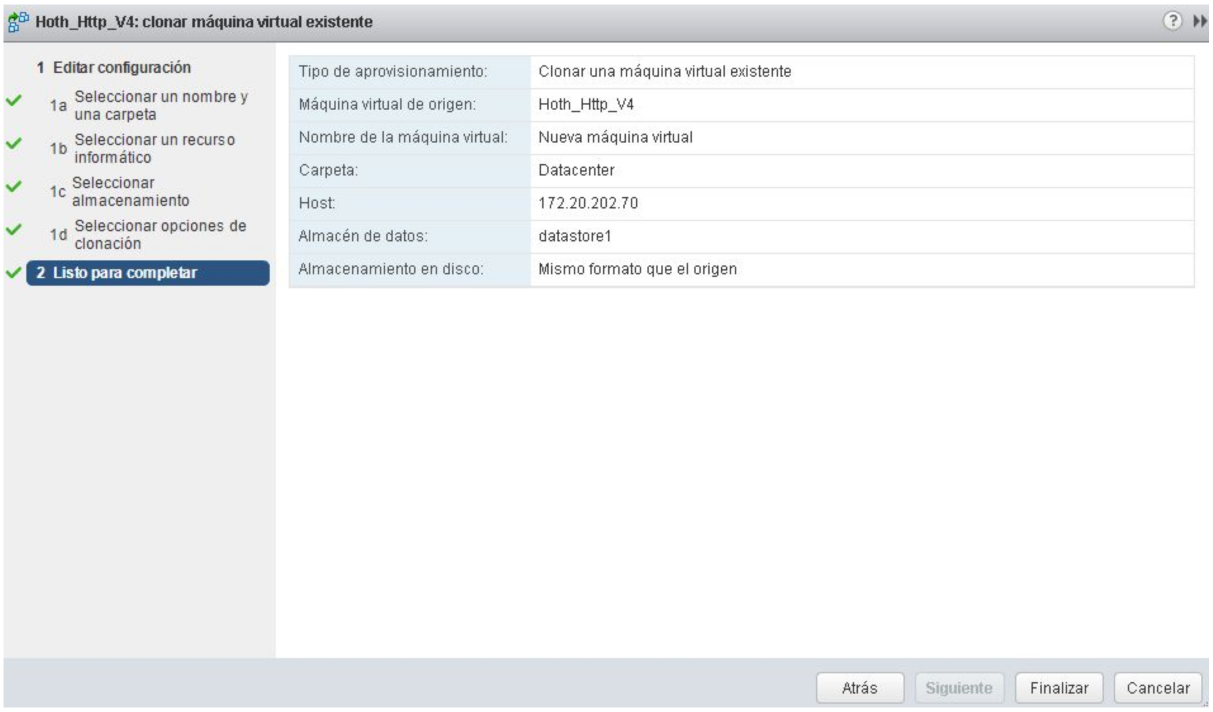
Buscar
Datacenter
ClusterSW
172.20.202.100
172.20.202.70

Seleccione un clúster, un host, una vApp o un grupo de recursos para ejecutar esta máquina virtual.

Compatibilidad
Las comprobaciones de compatibilidad se completaron correctamente.

Atrás | Siguiente | Finalizar | Cancelar

Then click on “Siguiente” two more times and this last windows appears
Here we can see the details of the clone and if we click on “Finalizar” it will clone it.


Hoth_Http_V4: clonar máquina virtual existente
1 Editar configuración
1a Seleccionar un nombre y una carpeta
1b Seleccionar un recurso informático
1c Seleccionar almacenamiento
1d Seleccionar opciones de clonación
2 Listo para completar
Atrás
Siguiente
Finalizar
Cancelar

Tipo de aprovisionamiento:	Clonar una máquina virtual existente
Máquina virtual de origen:	Hoth_Http_V4
Nombre de la máquina virtual:	Nueva máquina virtual
Carpeta:	Datacenter
Host:	172.20.202.70
Almacén de datos:	datastore1
Almacenamiento en disco:	Mismo formato que el origen

2.2.2-Corosync and pacemaker

Corosync is an open source program that provides cluster membership and messaging capabilities, often referred to as the messaging layer, to client servers. Pacemaker is an open source cluster resource manager (CRM), a system that coordinates resources and services that are managed and made highly available by a cluster. In essence, Corosync enables servers to communicate as a cluster, while Pacemaker provides the ability to control how the cluster behaves.

The first step is to install both corosync and pacemaker, we can use this command and the installation is going to be a little bit longer than usual.

```
apt-get install pacemaker
```

* Note that Corosync is installed as a dependency of the Pacemaker package.

On both servers:

```
chown -R hacluster:haclient /var/lib/heartbeat
```

Give to the user and group that has just been created the property of this folder so that we can execute comprobations in a secure space.

On the primary server, run the corosync-keygen script in order to create a 128-byte cluster authorization key, and write it to /etc/corosync/authkey.

```
corosync-keygen
```

On the primary server, copy the authkey to the secondary server into /etc/corosync/authkey folder.

```
scp /etc/corosync/authkey root@172.20.202.195:/etc/corosync/authkey
```

Restrict permissions to root:

```
chown root: /etc/corosync/authkey
chmod 400 /etc/corosync/authkey
```

Configure Corosync Cluster

In order to get our desired cluster up and running, we must set up corosync properly on both server. For this, we are going to edit the corosync configuration file using the file editor “nano”.

```
nanoi /etc/corosync/corosync.conf
```

Here is an example of a Corosync configuration file that will allow your servers to communicate as a cluster. bindnetaddr should be set to the private IP address of the server you are currently working on. The two other highlighted items should be set to the indicated server’s private IP address. With the exception of the bindnetaddr, the file should be identical on both servers.

Replace the contents of corosync.conf with this configuration, with the changes that are specific to your environment:
This is the whole corosync.conf file used for our cluster

The text in red is the part of the file that we have to edit in order to work with our personal settings and network.

For example in the first part, where the interface is set, we just have to copy it and change the “bindnetaddr” value to our network’s address.

And the second part is to specify the two nodes and the cluster is going to be consisted of, in our case, a machine with ip 172.20.202.199 and another one (the clone) with ip 172.20.202.195. Once again the only part that is needed to change is the “ring0_addr” with the ip of each machine and if we want, the name of the node.

corosync.conf

```
# Please read the corosync.conf.5 manual page
totem {
        version: 2

        # Corosync itself works without a cluster name, but DLM needs one.
        # The cluster name is also written into the VG metadata of newly
        # created shared LVM volume groups, if lvmlockd uses DLM locking.
        cluster_name: debian
        interface {
            ringnumber: 0
            bindnetaddr: 172.20.202.0
            broadcast: yes
            mcastport: 5405
  }
        # crypto_cipher and crypto_hash: Used for mutual node authentication.
        # If you choose to enable this, then do remember to create a shared
        # secret with "corosync-keygen".
        # enabling crypto_cipher, requires also enabling of crypto_hash.
        # crypto works only with knet transport
        crypto_cipher: none
        crypto_hash: none
}

logging {
        # Log the source file and line where messages are being
        # generated. When in doubt, leave off. Potentially useful for
        # debugging.
        fileline: off
        # Log to standard error. When in doubt, set to yes. Useful when
        # running in the foreground (when invoking "corosync -f")
        to_stderr: yes
        # Log to a log file. When set to "no", the "logfile" option
        # must not be set.
        to_logfile: yes
        logfile: /var/log/corosync/corosync.log
        # Log to the system log daemon. When in doubt, set to yes.
        to_syslog: yes
        # Log debug messages (very verbose). When in doubt, leave off.
        debug: off
        # Log messages with time stamps. When in doubt, set to hires (or on)
        #timestamp: hires
        logger_subsys {
                subsys: QUORUM
                debug: off
        }
}

quorum {
        # Enable and configure quorum subsystem (default: off)
        # see also corosync.conf.5 and votequorum.5
        provider: corosync_votequorum
}

nodelist {
        # Change/uncomment/add node sections to match cluster configuration

        node {
                # Hostname of the node
                name: node1
                # Cluster membership node identifier
                nodeid: 1
                # Address of first link
                ring0_addr: 172.20.202.199
                # When knet transport is used it's possible to define up to 8 links
                #ring1_addr: 192.168.1.1
        }
        # ...
        node {
                # Hostname of the node
                name: node2
                # Cluster membership node identifier
                nodeid: 2
                # Address of first link
                ring0_addr: 172.20.202.195
                # When knet transport is used it's possible to define up to 8 links
#ring1_addr: 192.168.1.1
        }
}
```

Once we have configured this file Pacemaker will be able to use Corosync to communicate with our servers.

But, if we want corosync to work, we first need to enable it, as it comes disabled by default, for this matter, we are going to edit a file in both machines, and add a line at the end of it.

nano /etc/default/corosync

Here we will just add the line “START=yes”

```
# Command line options
#OPTIONS=""
START=yes
```

After adding this line, the service is enabled, and we just need to start it:

service corosync start

Once Corosync is running on both servers, they should be clustered together. We can verify this by running this command:

corosync-cmapctl | grep members

The output of the command should look something like this:

```
runtime.members.1.config_version (u64) = 0
runtime.members.1.ip (str) = r(0) ip(172.20.202.199)
runtime.members.1.join_count (u32) = 1
runtime.members.1.status (str) = joined
runtime.members.2.config_version (u64) = 0
runtime.members.2.ip (str) = r(0) ip(172.20.202.195)
runtime.members.2.join_count (u32) = 1
runtime.members.2.status (str) = joined
```

We can see that both nodes have joined the cluster.
Now it’s time to configure pacemaker

Start and Configure Pacemaker

Pacemaker, which depends on the messaging capabilities of Corosync, is now ready to be started and to have its basic properties configured.
The Pacemaker service requires Corosync to be running, so it is disabled by default. In order to enable Pacemaker and to start on system boot we can use this command:

update-rc.d pacemaker defaults 20 01

With the prior command, we set Pacemaker's start priority to 20. It is important to specify a start priority that is higher than Corosync's (which is 19 by default), so that Pacemaker starts after Corosync.

After this command, we start pacemaker service:

service pacemaker start

To interact with Pacemaker, we can use the crm utility.
We first need to install this utility, with this command:

apt install crmsh

In order to check the status of Pacemaker we can use this crm command, it's quite simple.:

sudo crm status

The output should look like this:
As you can see, there is no resource configured, but there are two nodes configured, the ones we set in the corosync configuration. In our case both nodes are online, as we can see at the bottom of the output, and we can also see the names. If one of your nodes doesn't appear as "online" try waiting 30 seconds and checking the status again.

```
Current DC: node2 (version 2.0.1-9e909a5bdd) - partition with quorum
Last updated: Tue Jan 21 14:10:56 2020
Last change: Fri Jan 17 12:50:56 2020 by root via cibadmin on node2

2 nodes configured
0 resource configured

Online: [ node1 node2 ]
```

Configure Cluster Properties

Now we're ready to configure the basic properties of Pacemaker. Note that all Pacemaker (crm) commands can be run from either node server, as it automatically synchronizes all cluster-related changes across all member nodes.

For our desired setup, we want to disable STONITH—a mode that many clusters use to remove faulty nodes—because we are setting up a two-node cluster. To do so, run this command on either server:

```
crm configure property stonith-enabled=false
```

We also want to disable quorum-related messages in the logs:

```
crm configure property no-quorum-policy=ignore
```

Again, this setting only applies to 2-node clusters.

The next step is to install “curl” utility, as we are going to need it in order to donwload some files from a url.

```
apt-get install curl
```

Download FloatIP OCF Resource Agent

Pacemaker allows the addition of OCF resource agents by placing them in a specific directory.
On both servers, create the digitalocean resource agent provider directory with this command:

```
mkdir /usr/lib/ocf/resource.d/hoth
```

On both servers, download the FloatIP OCF Resource Agent:

```
curl -o /usr/lib/ocf/resource.d/hoth/floatip
https://gist.githubusercontent.com/thisismitch/b4c91438e56bfe6b7bfb/ra
w/2dffe2ae52ba2df575baae46338c155adbaef678/floatip-ocf
```

Them, make it executable:

```
chmod +x /usr/lib/ocf/resource.d/hoth/floatip
```

Use this command:

```
crm configure primitive clusterhoth ocf:heartbeat:IPaddr2 params
ip="172.20.202.190" cidr_netmask="24" op monitor interval="30s"
```

Should be done, now check the status

crm status

It should look like this:

```
Stack: corosync
Current DC: node2 (version 2.0.1-9e909a5bdd) - partition with quorum
Last updated: Tue Jan 21 14:25:54 2020
Last change: Fri Jan 17 12:50:56 2020 by root via cibadmin on node2

2 nodes configured
1 resource configured

Online: [ node1 node2 ]

Full list of resources:

 clusterhoth    (ocf::heartbeat:IPaddr2):       Started node2
```

3-Streaming

3.1-Introduction

Some rebel allies have created a series of multimedia content (audio and videos) explaining some questions, such as the maintenance of your X-Wing or how to disable imperial base and spaceship weapons. Due to having this media available, we want to create a Media Center that offers all this content in **on-demand Streaming** using Kodi. It must be configured in order to be accessible through UPnP and to be manageable through a web browser.

Years ago, clients had to download the file completely before playing it back. This all changed with **streaming video.** Content is served in a way that allows files to play almost immediately after the file begins to download. Streaming video technology is harder to copy and prevents users from saving a copy to their computer if you don't want them to.

We have selected the **Kodi media center** for this purpose as its compatible with UPnP protocol.

Universal Plug and Play (UPnP) is a set of networking protocols that permits networked devices, such as personal computers, printers, Internet gateways, Wi-Fi access points and mobile devices to seamlessly discover each other's presence on the network and establish functional network services for data sharing, communications, and entertainment.

For the checking of the UPnP protocol, we are going to use **Windows VLC Media Player.**

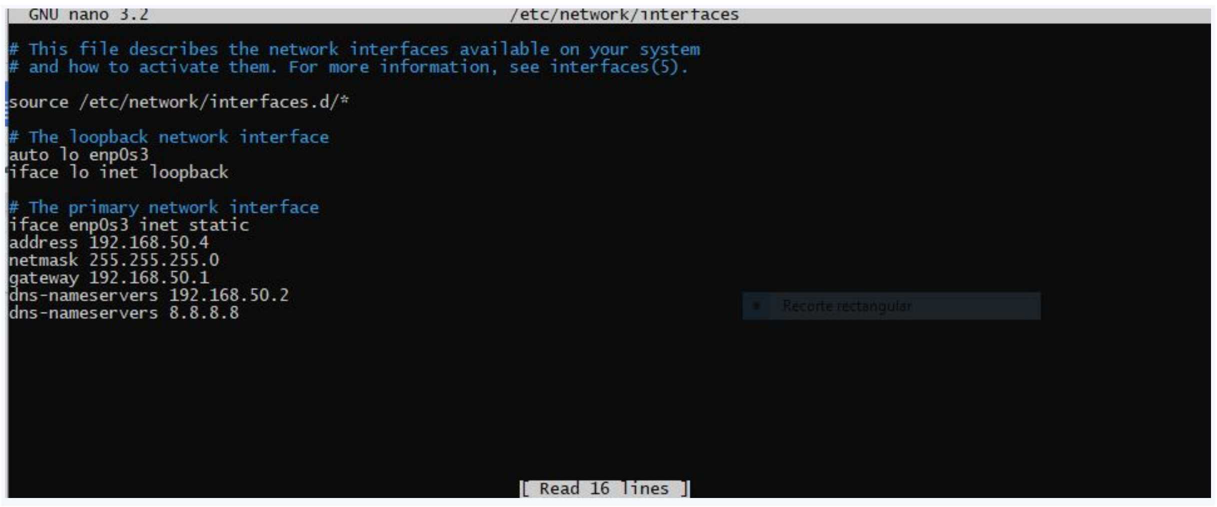
In order to simulate a streaming video conference, there are many programs to use such as **Skype** or **Discord** but as we don't have webcams in our computer we are going to use **TeamSpeak.**

3.2-Configuration

Depending on which operating system you are working with you will have to install Kodi in different ways. In our bibliography there is a link you may want to check in order to install Kodi properly. In case you have a Debian 10, check this configuration.

First of all we have to ensure that the network configuration is working. Here is a screenshot of our configuration. Don't forget to restart the network service after making any kind of changes in this file:

```
nano /etc/network/interfaces
```



```
  GNU nano 3.2                         /etc/network/interfaces

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

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo enp0s3
iface lo inet loopback

# The primary network interface
iface enp0s3 inet static
address 192.168.50.4
netmask 255.255.255.0
gateway 192.168.50.1
dns-nameservers 192.168.50.2
dns-nameservers 8.8.8.8

                                 [ Read 16 lines ]
```

```
service networking restart
```

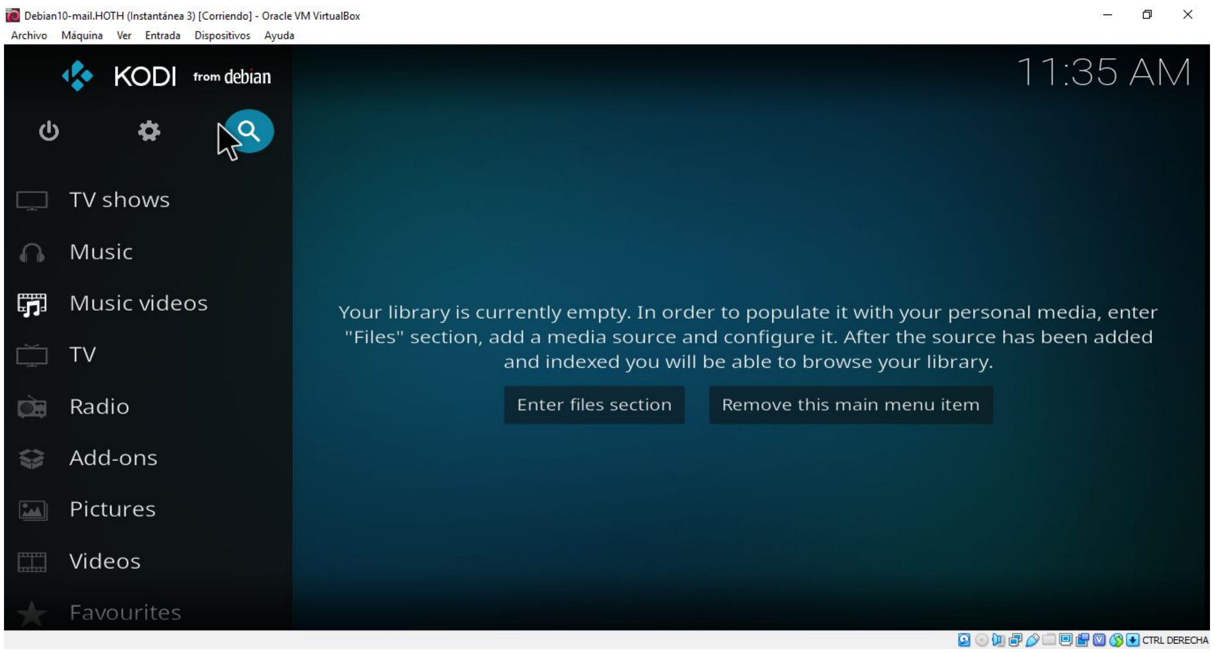
For the correct management os Kodi is recommendable to install a graphical environment. In our case we are going to install Gnome.

```
apt-get install gnome
```

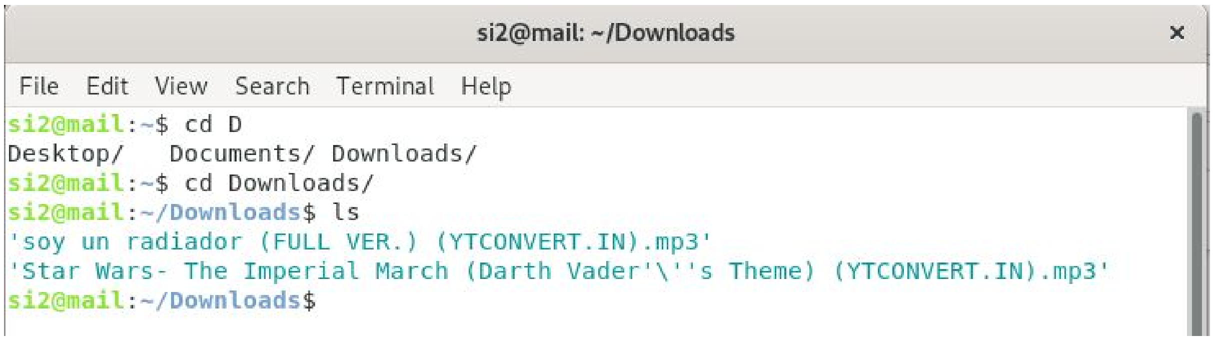
```
sudo apt install kodi
```

```
apt update
```

After the installation, we have downloaded two .mp3 files to show you how it works. This files are going to be stored by default in the users Downloads directory. You may want to create a direct link to this media source in order to be able to browse the library.


Debian10-mail.HOTH (Instantánea 3) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
KODI from debian
11:35 AM
TV shows
Music
Music videos
TV
Radio
Add-ons
Pictures
Videos
Favourites
Enter files section
Remove this main menu item
CTRL DERECHA

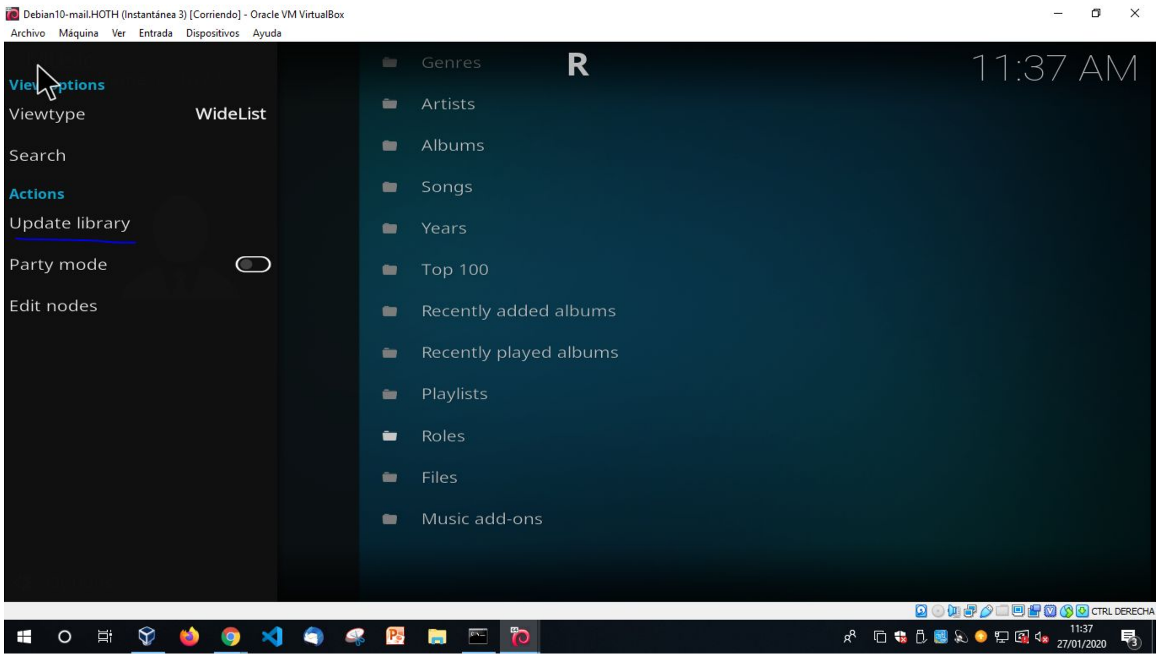
Your library is currently empty. In order to populate it with your personal media, enter "Files" section, add a media source and configure it. After the source has been added and indexed you will be able to browse your library.



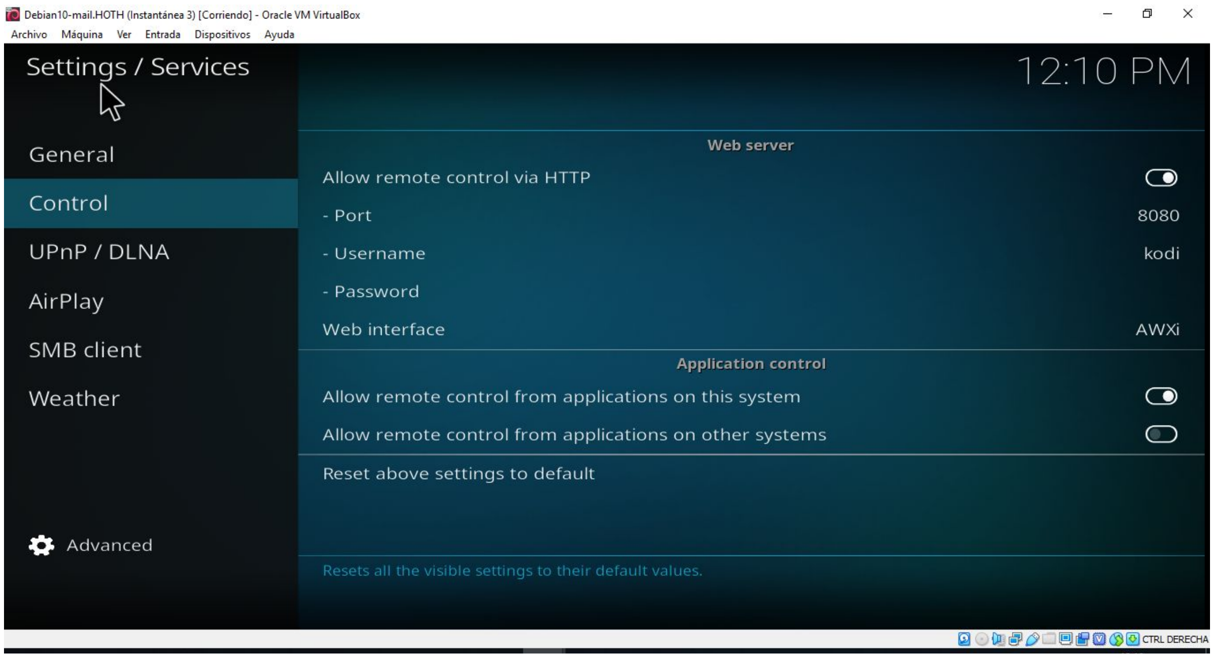
```
si2@mail:~$ cd D
Desktop/   Documents/ Downloads/
si2@mail:~$ cd Downloads/
si2@mail:~/Downloads$ ls
'soy un radiador (FULL VER.) (YTCONVERT.IN).mp3'
'Star Wars- The Imperial March (Darth Vader'\''s Theme) (YTCONVERT.IN).mp3'
si2@mail:~/Downloads$
```

In case the files are not shown there are many ways to update the library, for example moving the mouse to the left and an option menu will be shown with this option.

After updating the library, there are going to appear some categories which are created by Kodi.



Kodi also offers the possibility of managing the files via web browser. By default the port is 8080 and you can also set an username and password for more security. Web interface can also be changed.


Debian10-mail.HOTH (Instantánea 3) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Settings / Services
12:10 PM
General
Control
UPnP / DLNA
AirPlay
SMB client
Weather
Advanced

Web server

Allow remote control via HTTP

- Port 8080

- Username kodi

- Password

Web interface AWXi

Application control

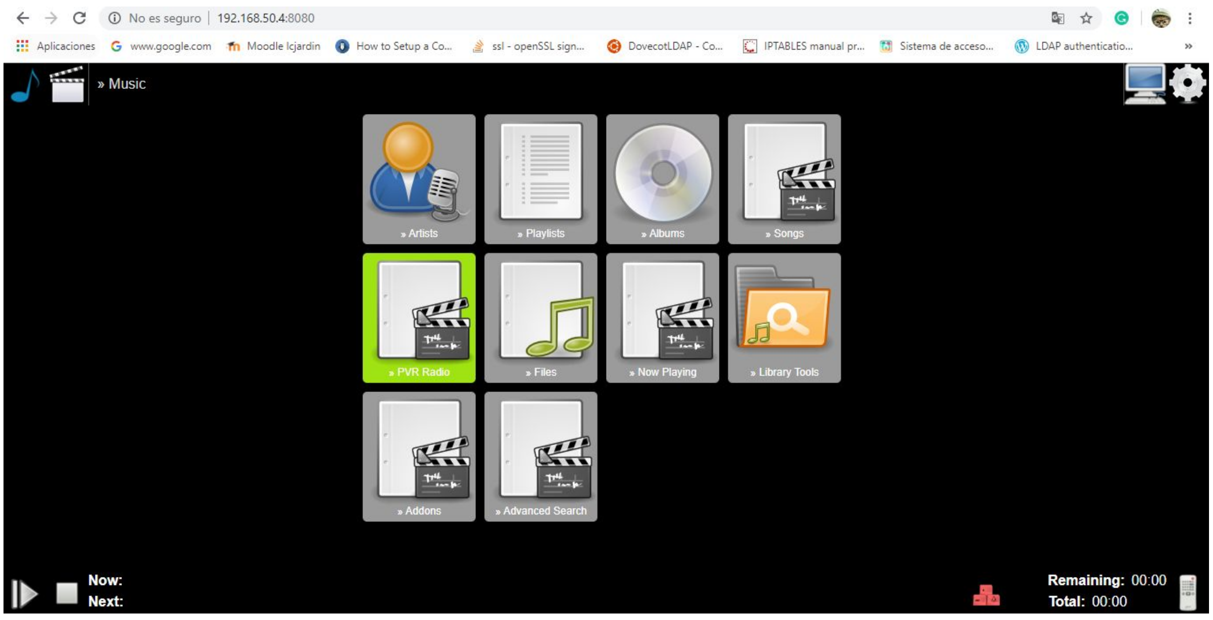
Allow remote control from applications on this system

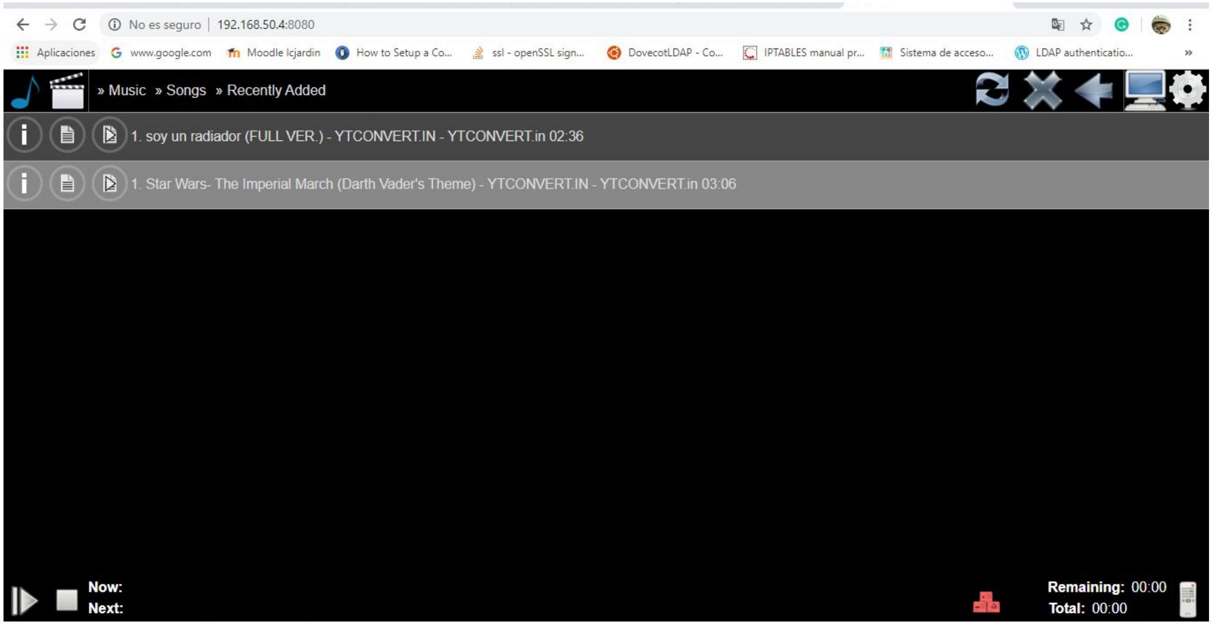
Allow remote control from applications on other systems

Reset above settings to default

Resets all the visible settings to their default values.

Julen Fernandez
Ibon Pina
SI2

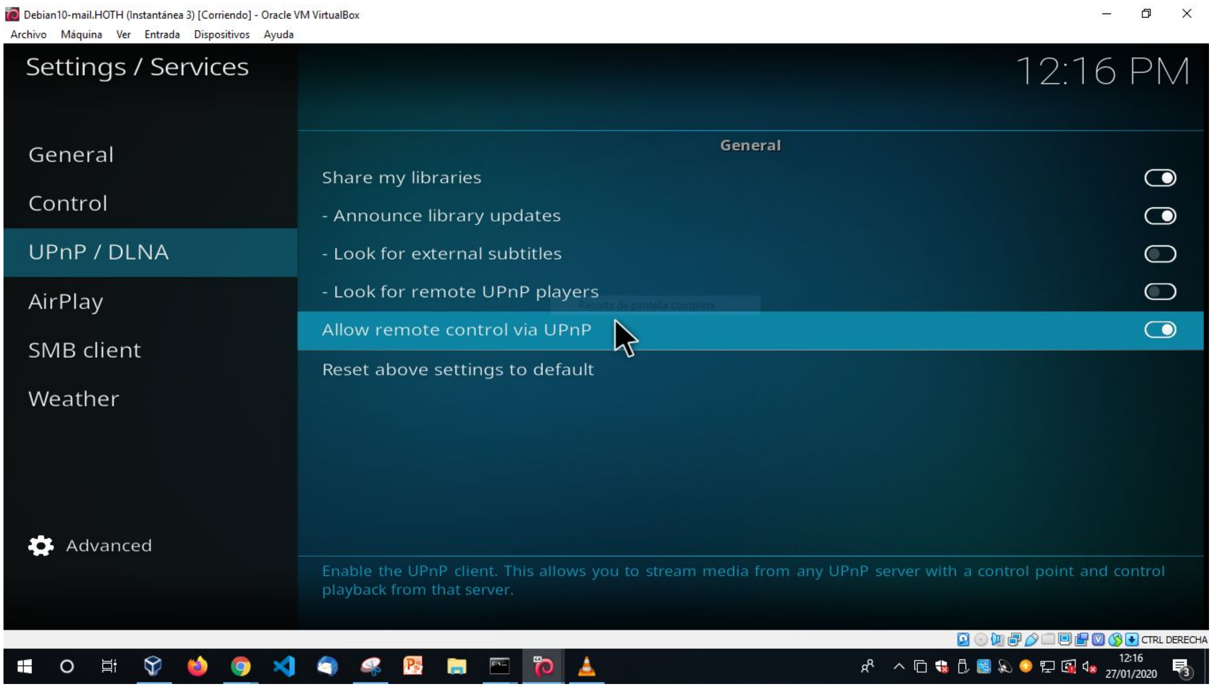



No es seguro | 192.168.50.4:8080
» Music » Songs » Recently Added
Now:
Next:
Remaining: 00:00
Total: 00:00

1. soy un radiador (FULL VER.) - YTCONVERT.IN - YTCONVERT.in 02:36
1. Star Wars- The Imperial March (Darth Vader's Theme) - YTCONVERT.IN - YTCONVERT.in 03:06

Kodi is compatible with UPnP protocol and it is very easy to use and activate it.

Julen Fernandez
Ibon Pina
SI2


Settings / Services

12:16 PM

General

Share my libraries: on

- Announce library updates: on

- Look for external subtitles: off

- Look for remote UPnP players: off

Allow remote control via UPnP: on

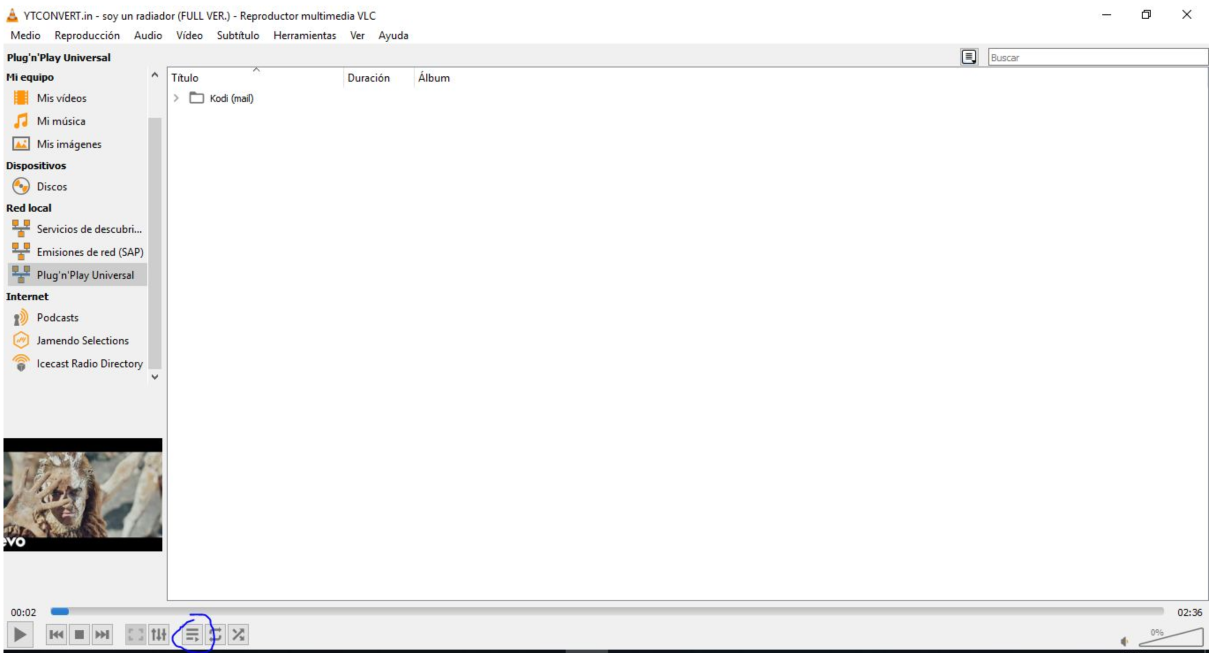
Reset above settings to default

Enable the UPnP client. This allows you to stream media from any UPnP server with a control point and control playback from that server.

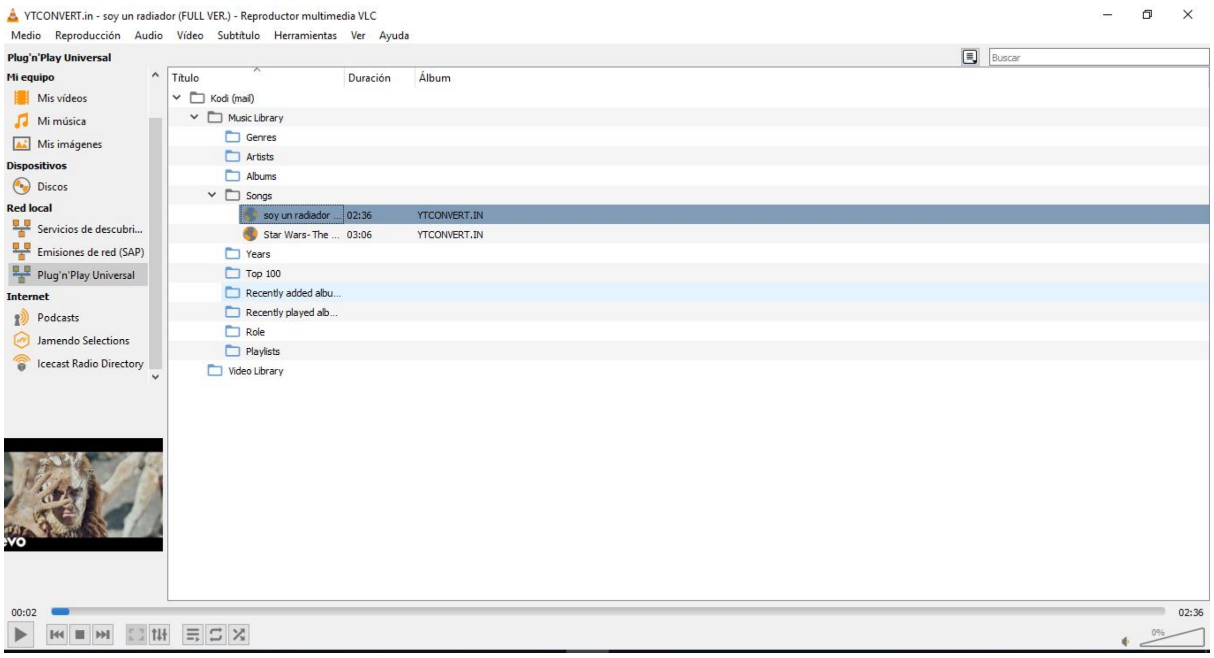
Debian10-mail.HOTH (Instantánea 3) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
General
Control
UPnP / DLNA
AirPlay
SMB client
Weather
Advanced
CTRL DERECHA
12:16
27/01/2020

For the checking that the files are shared, we are going to use Windows VLC Media Player.

We have to open the library, clicking the rounded out icon, and we click on the local networks UpnP option. There must appear our Kodi server with each correspondent categories and files.



Julen Fernandez
Ibon Pina
SI2



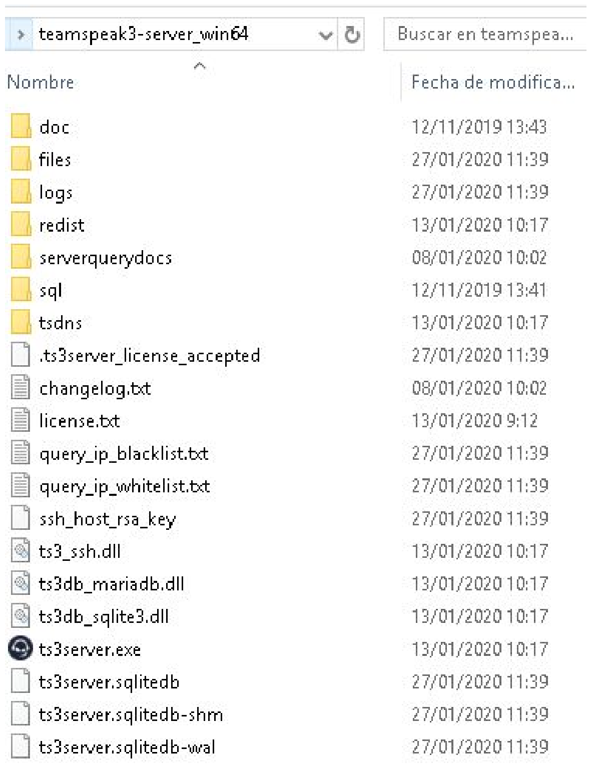
3.3-Video conference streaming

One of the aims of this challenge is to communicate with each other inside our planet network, we already have voip, but we also thought on using more communication options such as discord or teamspeak.

Using discord is very easy, you just need to enter their webpage and choose and alias, then add your mate and call him, all in a web interface.

But we also tried using teamspeak, you only need to download the client for your operating system, install it and choose a nickname, but unlike discord, you can’t just call a friend, in this case, the only way to chat with someone is joining a teamspeak server. This seems easy, but the main problem is that we can’t access any public server from our class network. That’s why we have created our own server downloading some files from the official teamspeak web page.

Server files:

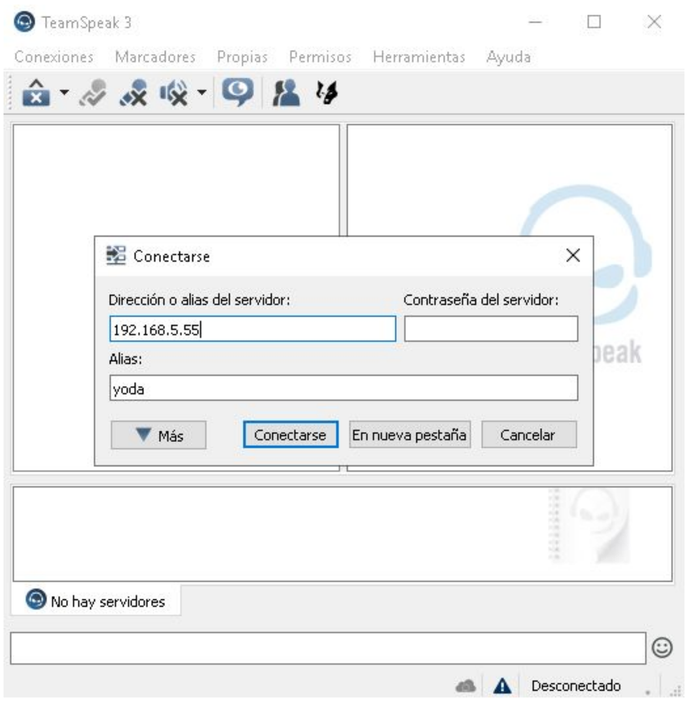


teamspeak3-server_win64

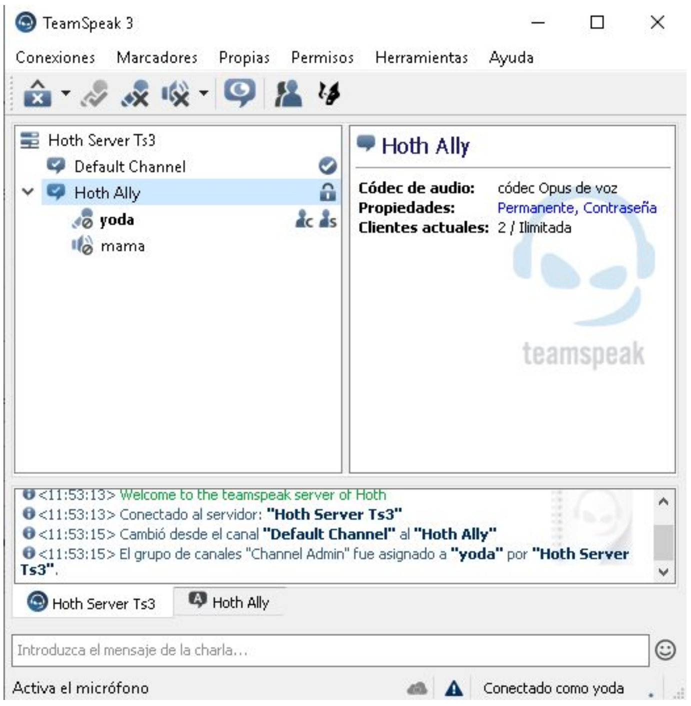
Buscar en teamspea...

Nombre	Fecha de modifica...
doc	12/11/2019 13:43
files	27/01/2020 11:39
logs	27/01/2020 11:39
redist	13/01/2020 10:17
serverquerydocs	08/01/2020 10:02
sql	12/11/2019 13:41
tsdns	13/01/2020 10:17
.ts3server_license_accepted	27/01/2020 11:39
changelog.txt	08/01/2020 10:02
license.txt	13/01/2020 9:12
query_ip_blacklist.txt	27/01/2020 11:39
query_ip_whitelist.txt	27/01/2020 11:39
ssh_host_rsa_key	27/01/2020 11:39
ts3_ssh.dll	13/01/2020 10:17
ts3db_mariadb.dll	13/01/2020 10:17
ts3db_sqlite3.dll	13/01/2020 10:17
ts3server.exe	13/01/2020 10:17
ts3server.sqlitedb	27/01/2020 11:39
ts3server.sqlitedb-shm	27/01/2020 11:39
ts3server.sqlitedb-wal	27/01/2020 11:39

There is no need to edit any configuration file, you just decompress it and start the .exe file.
Then you join the server from the teamspeak client using the ip of it, in this case 192.168.5.55
and you type a password



Once you are connected to the server, just join any channel and talk with anyone.



Códec de audio: códec Opus de voz
Propiedades: Permanente, Contraseña
Clientes actuales: 2 / Ilimitada

```
<11:53:13> Welcome to the teamspeak server of Hoth
<11:53:13> Conectado al servidor: "Hoth Server Ts3"
<11:53:15> Cambió desde el canal "Default Channel" al "Hoth Ally"
<11:53:15> El grupo de canales "Channel Admin" fue asignado a "yoda" por "Hoth Server Ts3".
```

3.4-Firewall Configuration

```
iptables -A FORWARD -p tcp --dport 8080 -j ACCEPT && echo "  FORWARD: 8080 (Kodi)"
```

```
iptables -t nat -A PREROUTING -i $IFACE_OUT -p tcp --dport 8066 -j DNAT --to $IP_MAIL:8080
```

After adding these lines we must execute the script again.

4-VOIP

4.1-Introduction

We suspect that Galactic Empire spies are catching our communications, so we have decided to change this communication systems. Voice communications that have been done through radio must be substituted by **Voice IP communications** (Asterisk 13 under FreePBX): each user must have an associated extension to which anyone can call, but the option to call an entire group (Jedi Masters, Generals, Captains or Pilots) will be possible as well, the music on hold must be personalized and the calls to Jedi Masters, if they are not in their extension, must call to all the extensions. In order to substitute holographic technology communications, we must find some tools that permit us **Live Streaming** communications, with, more or less, the same features that the VoIP server has.

Voice communication is necessary for our system so that all our planets can communicate with each other with no troubles and the enemies are not going to expect this change.

Asterisk is a software implementation, based on Linux, of a private branch exchange (PBX). In conjunction with suitable telephony hardware interfaces and network applications, Asterisk is used to establish and control telephone calls between telecommunication endpoints.

VOIP is the selected technology for this purpose. VoIP is a set of rules, devices, and protocols that allows communicating voice over IP protocol.

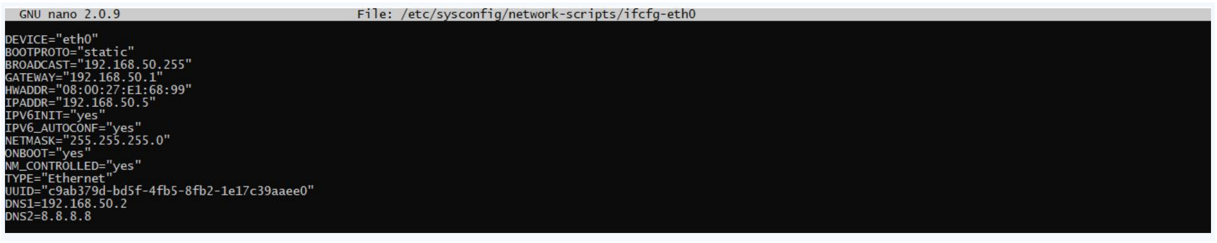
The **Session Initiation Protocol** (**SIP**) is a signaling protocol used for initiating, maintaining, and terminating real-time sessions that include voice, video and messaging applications.

Zoiper is going to be the program used for the checkings. In the webgraphy there is a link for its download.

4.2-Configuration

First of all we have to ensure that the network configuration is working. Here is a screenshot of our configuration. Don't forget to restart the network service (Or whole computer) after making any kind of changes in this file:

```
nano /etc/sysconfig/network-scripts/ifcfg-eth0
```



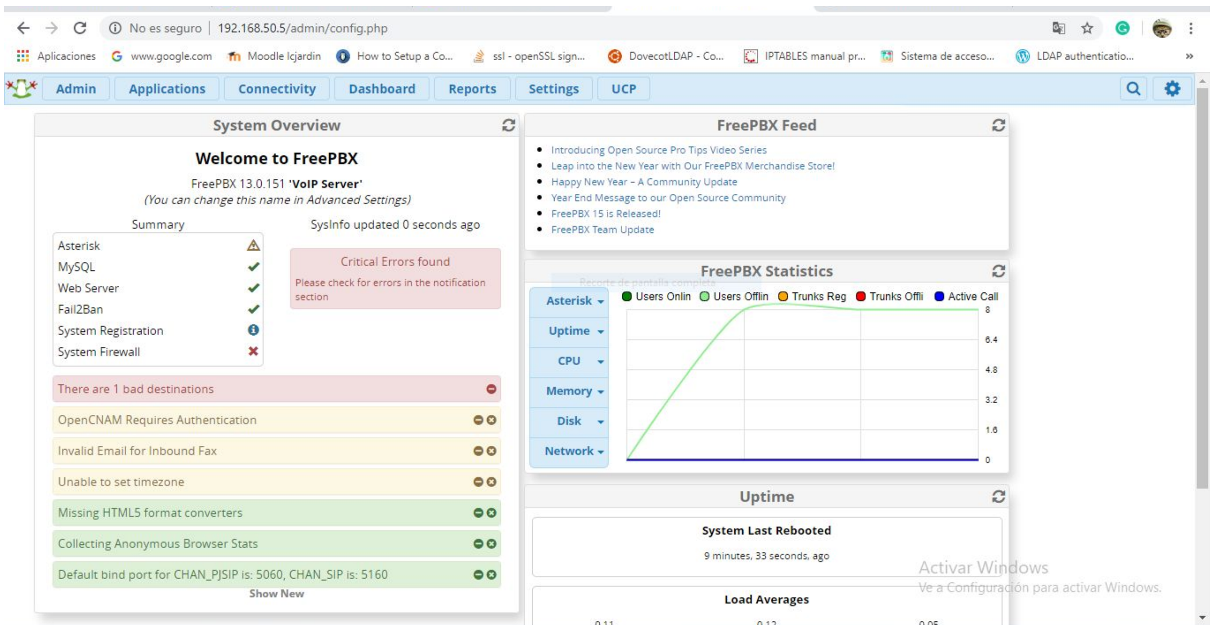
```
  GNU nano 2.0.9                          File: /etc/sysconfig/network-scripts/ifcfg-eth0

DEVICE="eth0"
BOOTPROTO="static"
BROADCAST="192.168.50.255"
GATEWAY="192.168.50.1"
HWADDR="08:00:27:E1:68:99"
IPADDR="192.168.50.5"
IPV6INIT="yes"
IPV6_AUTOCONF="yes"
NETMASK="255.255.255.0"
ONBOOT="yes"
NM_CONTROLLED="yes"
TYPE="Ethernet"
UUID="c9ab379d-bd5f-4fb5-8fb2-1e17c39aaee0"
DNS1=192.168.50.2
DNS2=8.8.8.8
```

```
reboot
```

All the management of the VoIP technology is going to be via web browser.

This is the main page that is shown when entering the web, typing the IP (192.168.50.5)in the web navigator.
In this managing webpage we should submit and apply all the changes before trying any kind of checking.



No es seguro | 192.168.50.5/admin/config.php

Admin | Applications | Connectivity | Dashboard | Reports | Settings | UCP

System Overview

Welcome to FreePBX

FreePBX 13.0.151 **'VoIP Server'**

(You can change this name in Advanced Settings)

Summary — SysInfo updated 0 seconds ago

Asterisk
MySQL ✓
Web Server ✓
Fail2Ban ✓
System Registration
System Firewall ✗

Critical Errors found
Please check for errors in the notification section

There are 1 bad destinations
OpenCNAM Requires Authentication
Invalid Email for Inbound Fax
Unable to set timezone
Missing HTML5 format converters
Collecting Anonymous Browser Stats
Default bind port for CHAN_PJSIP is: 5060, CHAN_SIP is: 5160
Show New

FreePBX Feed

- Introducing Open Source Pro Tips Video Series
- Leap into the New Year with Our FreePBX Merchandise Store!
- Happy New Year – A Community Update
- Year End Message to our Open Source Community
- FreePBX 15 is Released!
- FreePBX Team Update

FreePBX Statistics

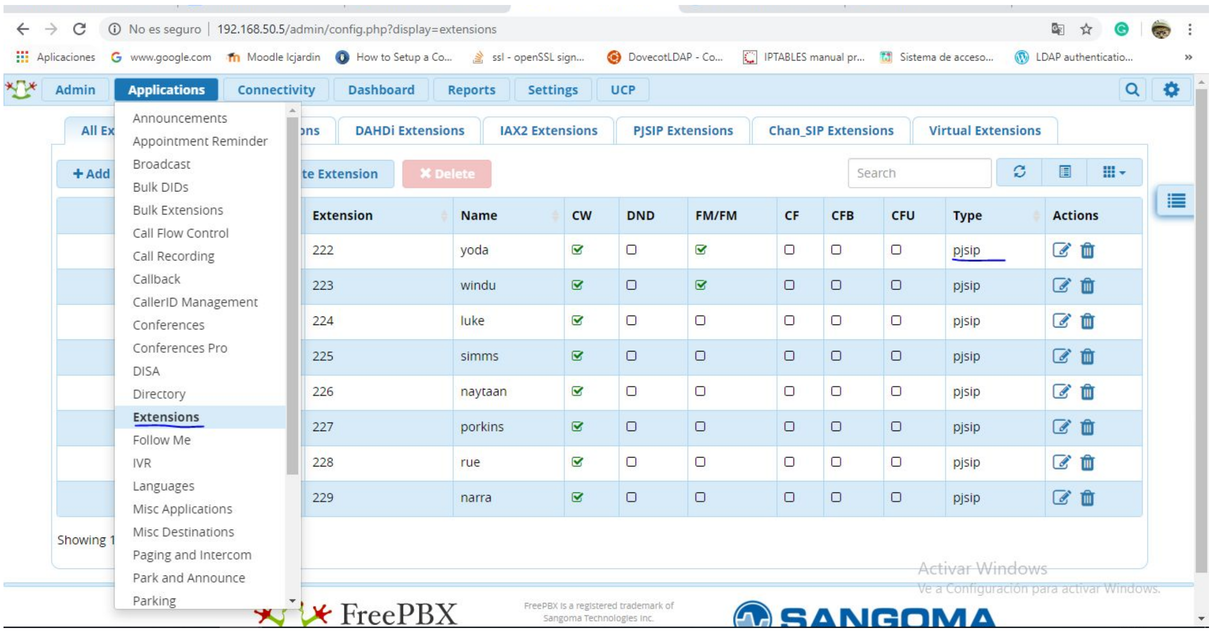
Uptime

System Last Rebooted
9 minutes, 33 seconds, ago

Load Averages

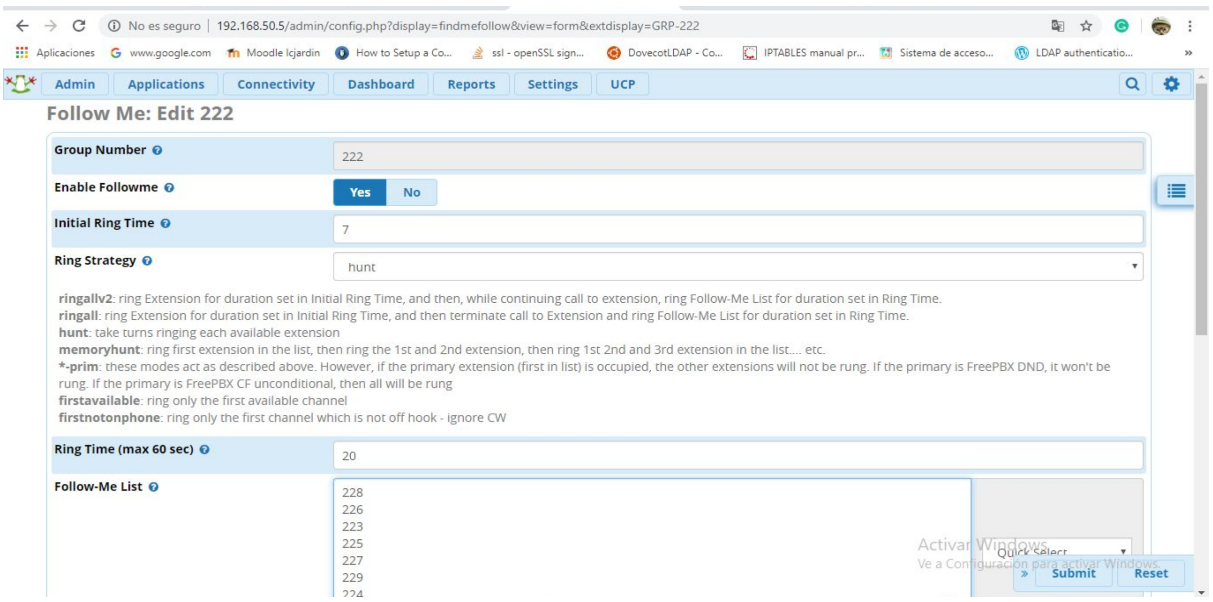
Activar Windows
Ve a Configuración para activar Windows.

Here we have to create all the necessary users with each correspondent extension and protocol.
The extension is going to be a numeric code and the protocol, the type, pjsip in our case. For each user you can add a password.



Extension	Name	CW	DND	FM/FM	CF	CFB	CFU	Type
222	yoda	✓	☐	✓	☐	☐	☐	pjsip
223	windu	✓	☐	✓	☐	☐	☐	pjsip
224	luke	✓	☐	☐	☐	☐	☐	pjsip
225	simms	✓	☐	☐	☐	☐	☐	pjsip
226	naytaan	✓	☐	☐	☐	☐	☐	pjsip
227	porkins	✓	☐	☐	☐	☐	☐	pjsip
228	rue	✓	☐	☐	☐	☐	☐	pjsip
229	narra	✓	☐	☐	☐	☐	☐	pjsip

For our Jedi Masters, Yoda and Windu, we have configured the follow me option, which consists on calling other extensions in case they dont answer the phone. There are many different options as shown in the next screenshot. We have chosen hunt, which after calling 7 seconds to our Jedi Masters and receiving no response, will continue ringing to the extensions one by one as selected in the follow-me list. Those extensions are going to receive the call for 20 seconds until it calls the next extension.


No es seguro | 192.168.50.5/admin/config.php?display=findmefollow&view=form&extdisplay=GRP-222
Admin
Applications
Connectivity
Dashboard
Reports
Settings
UCP
Quick Select
Submit
Reset

Follow Me: Edit 222

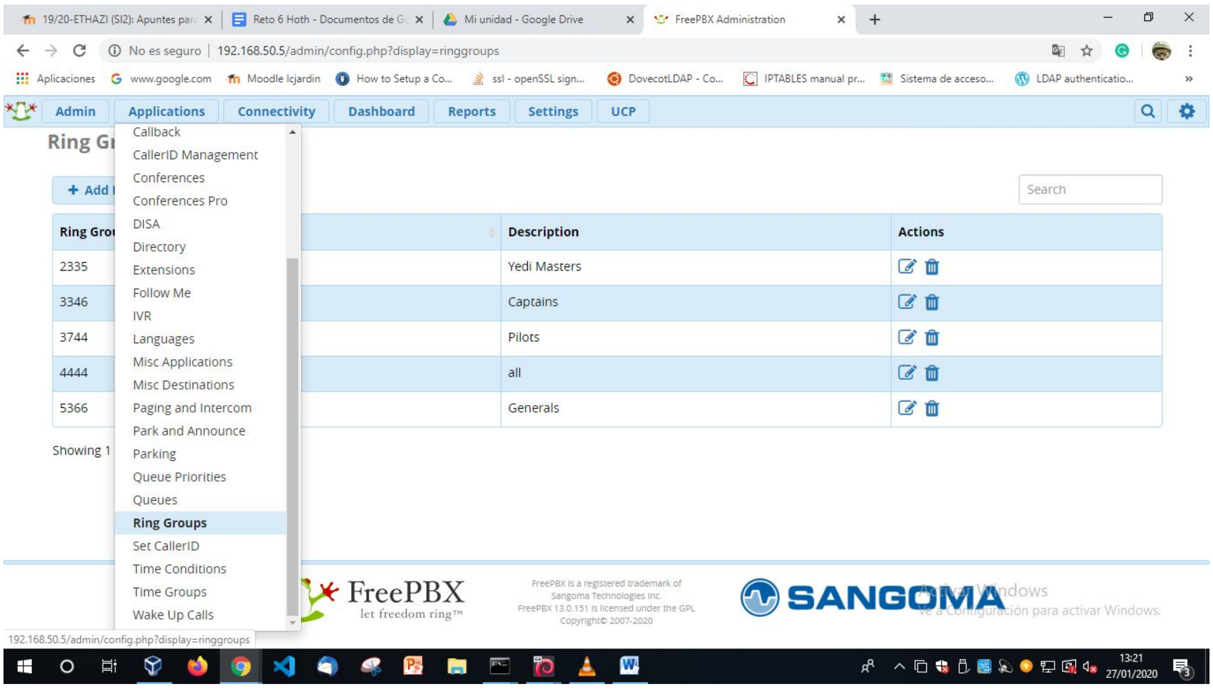
Group Number	222
Enable Followme	[✓ Yes] No
Initial Ring Time	7
Ring Strategy	hunt

ringallv2: ring Extension for duration set in Initial Ring Time, and then, while continuing call to extension, ring Follow-Me List for duration set in Ring Time.
ringall: ring Extension for duration set in Initial Ring Time, and then terminate call to Extension and ring Follow-Me List for duration set in Ring Time.
hunt: take turns ringing each available extension
memoryhunt: ring first extension in the list, then ring the 1st and 2nd extension, then ring 1st 2nd and 3rd extension in the list.... etc.
***-prim**: these modes act as described above. However, if the primary extension (first in list) is occupied, the other extensions will not be rung. If the primary is FreePBX DND, it won't be rung. If the primary is FreePBX CF unconditional, then all will be rung
firstavailable: ring only the first available channel
firstnotonphone: ring only the first channel which is not off hook - ignore CW

Ring Time (max 60 sec)	20
Follow-Me List	228 226 223 225 227 229 224

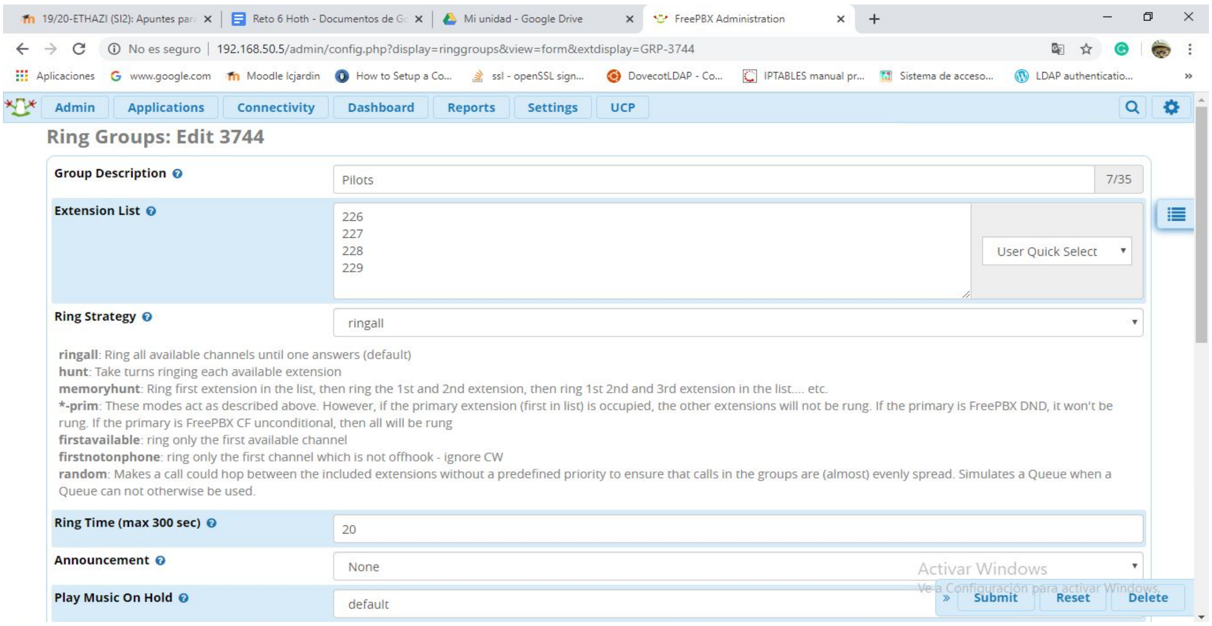
We have also created some ring groups, in order to call at the same time to more than one extension.

Julen Fernandez
Ibon Pina
SI2



192.168.50.5/admin/config.php?display=ringgroups

Ring Gro	Description	Actions
2335	Yedi Masters	
3346	Captains	
3744	Pilots	
4444	all	
5366	Generals	


192.168.50.5/admin/config.php?display=ringgroups&view=form&extdisplay=GRP-3744

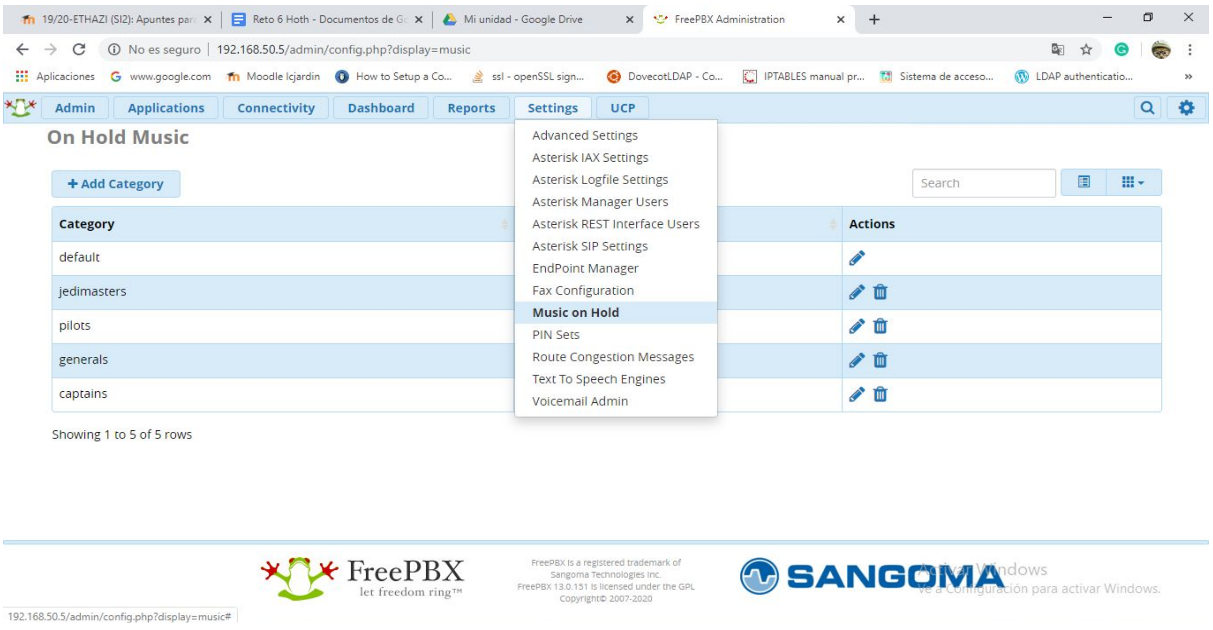
Ring Groups: Edit 3744

Group Description	Pilots (7/35)
Extension List	226 227 228 229
Ring Strategy	ringall

ringall: Ring all available channels until one answers (default)
hunt: Take turns ringing each available extension
memoryhunt: Ring first extension in the list, then ring the 1st and 2nd extension, then ring 1st 2nd and 3rd extension in the list.... etc.
***-prim**: These modes act as described above. However, if the primary extension (first in list) is occupied, the other extensions will not be rung. If the primary is FreePBX DND, it won't be rung. If the primary is FreePBX CF unconditional, then all will be rung
firstavailable: ring only the first available channel
firstnotonphone: ring only the first channel which is not offhook - ignore CW
random: Makes a call could hop between the included extensions without a predefined priority to ensure that calls in the groups are (almost) evenly spread. Simulates a Queue when a Queue can not otherwise be used.

Ring Time (max 300 sec)	20
Announcement	None
Play Music On Hold	default

Music on hold has also been configured. We can find this tool in Settings.



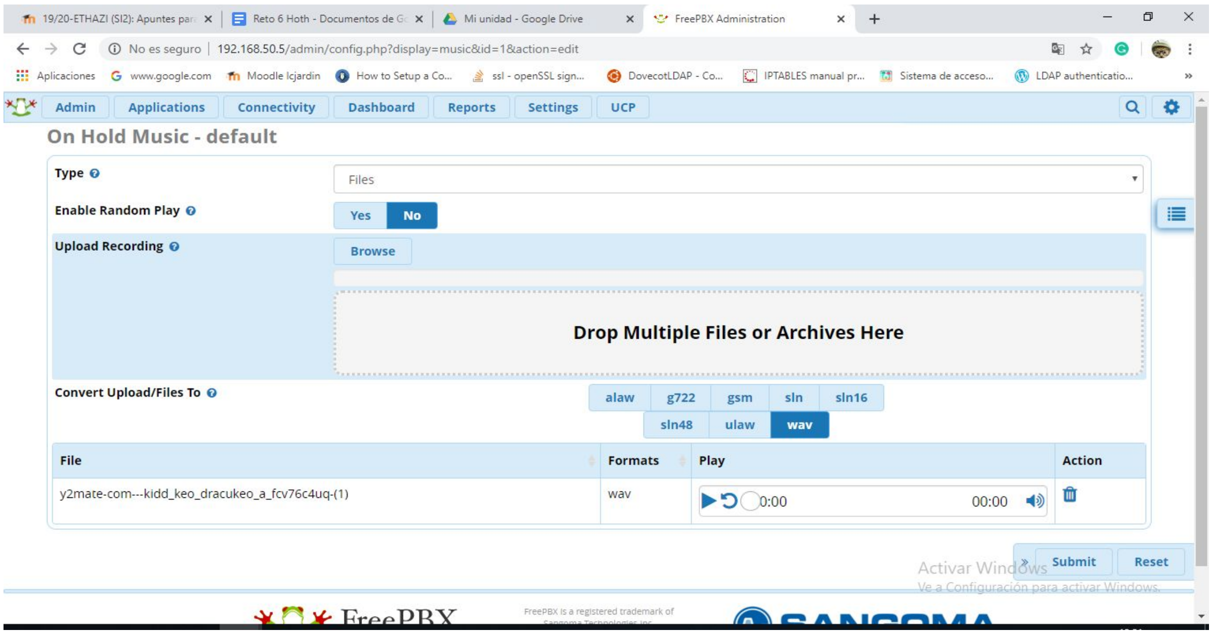
192.168.50.5/admin/config.php?display=music

On Hold Music

Category	Actions
default	✎
jedimasters	✎ 🗑
pilots	✎ 🗑
generals	✎ 🗑
captains	✎ 🗑

Showing 1 to 5 of 5 rows

The idea was to create a different one for each group but we finally decided to change the default one.



192.168.50.5/admin/config.php?display=music&id=1&action=edit

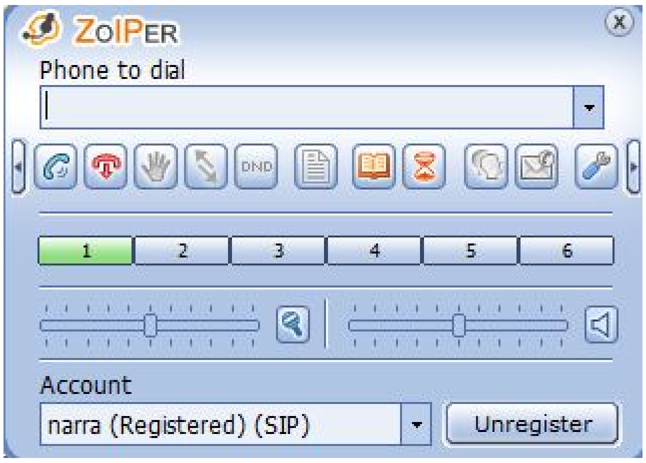
On Hold Music - default

Type	Files
Enable Random Play	Yes [✓ No]
Convert Upload/Files To	alaw g722 gsm sln sln16 sln48 ulaw [✓ wav]

File	Formats	Play	Action
y2mate-com---kidd_keo_dracukeo_a_fcv76c4uq-(1)	wav	0:00 00:00	🗑

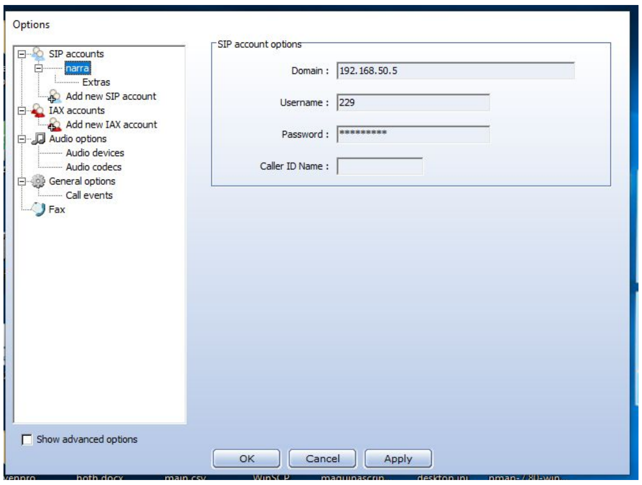
4.2.1-Checkings

The selected tool for the checkings is Zoiper Classic. To add a new user we must click in the icon on the far right.



narra (Registered) (SIP)

Here we have to add the domain, the extension number and a password if configured.



SIP account options

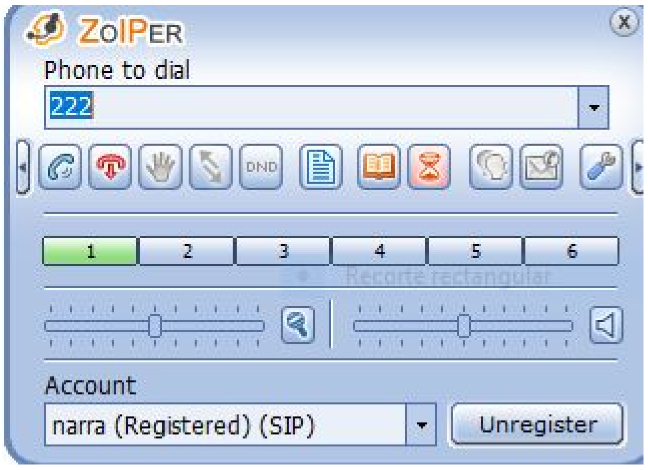
Domain : 192.168.50.5

Username : 229

Password : *********

Caller ID Name :

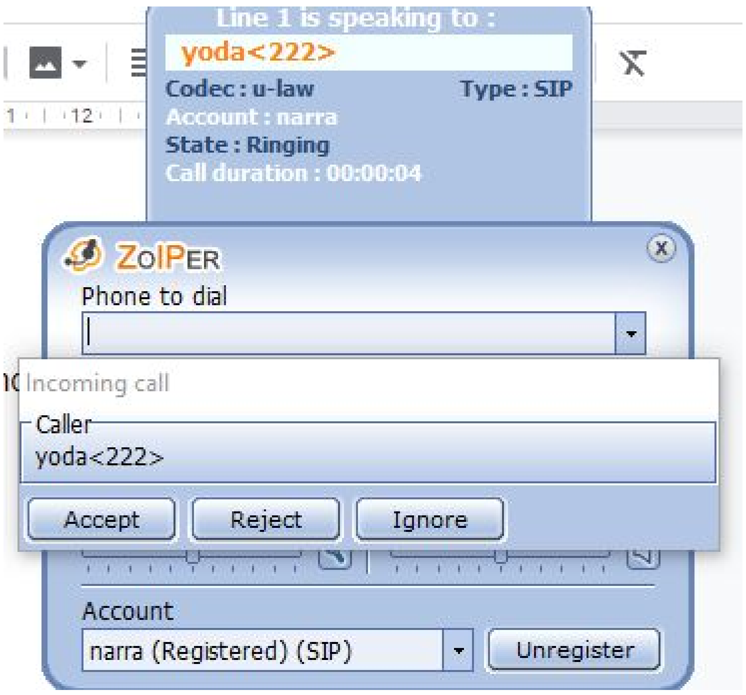
In order to call any user, we must call to the extension.



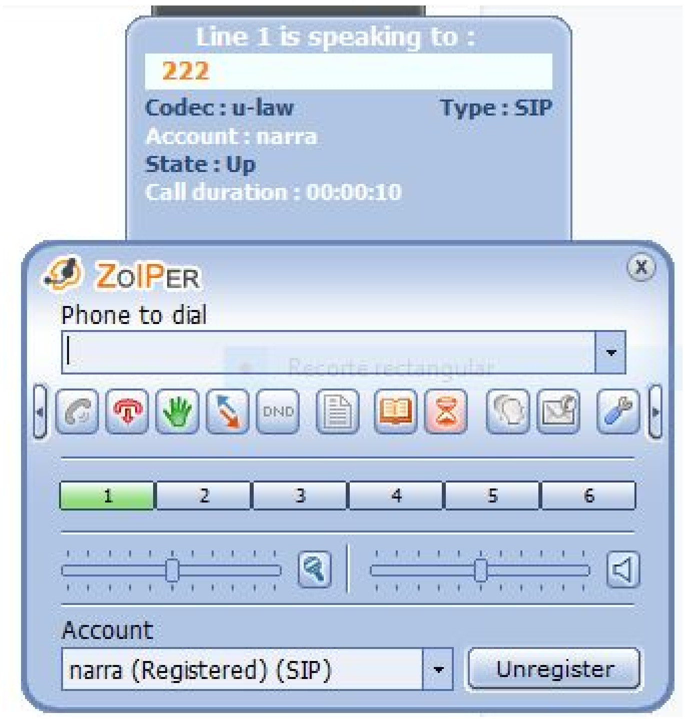
Phone to dial: 222

narra (Registered) (SIP)

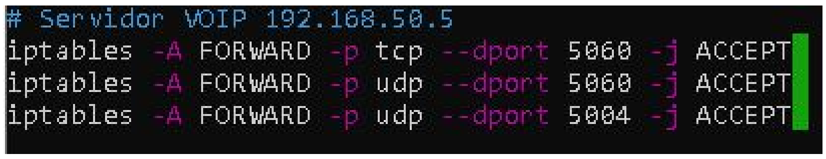
When receiving a call a message like this will appear.


Line 1 is speaking to :
yoda<222>
Codec : u-law
Type : SIP
Account : narra
State : Ringing
Call duration : 00:00:04

After picking up the phone, you will be able to hang out or leaving the call on hold. (The two far left icons, the red one for hanging out and the green one for leaving the call on hold.


Line 1 is speaking to :
222
Codec : u-law
Type : SIP
Account : narra
State : Up
Call duration : 00:00:10

4.3- Firewall Configuration



```
# Servidor VOIP 192.168.50.5
iptables -A FORWARD -p tcp --dport 5060 -j ACCEPT
iptables -A FORWARD -p udp --dport 5060 -j ACCEPT
iptables -A FORWARD -p udp --dport 5004 -j ACCEPT
```

After adding these lines we must execute the script again.

5-Webgraphy

Task scheduler
https://crontab.guru/
https://www.hostinger.es/tutoriales/como-usar-comando-tar-linux/
https://serverfault.com/questions/241588/how-to-automate-ssh-login-with-password
High availability
https://aula128.wordpress.com/2015/02/28/alta-disponibilidad-como-configurar-un-cluster-ha-linux-con-corosync-y-pacemaker-con-recurso-apache2/
https://www.digitalocean.com/community/tutorials/how-to-create-a-high-availability-setup-with-corosync-pacemaker-and-floating-ips-on-ubuntu-14-04
Streaming
https://kodi.wiki/view/Settings/Services/UPnP_DLNA
https://kodi.wiki/view/Web_interface
https://kodi.wiki/view/HOW-TO:Install_Kodi_for_Linux
VoIP
https://www.zoiper.com/en/voip-softphone/download/classic
https://www.microsip.org/