

Documentation Naboo



Naboo
Julen Fernández
Ibon Pina
SI2

Index

Cover	1
Index	2
Contextualization	3
Router configuration	6
Network configuration	6
SSH configuration	9
Webgraphy	11
Naboo.local configuration	12
Windows Server 2016	12
Domain	17
Users and Groups	18
DNS	20
Webgraphy	29
Raspberry-Pi	29
DHCP	31
Debian10 Client	34
Windows 10 Client	37
Webgraphy	38
Naboo.ally configuration	39
Linux DNS	39
Webgraphy	44
Issues during the configuration	45

Contextualization

For this project we are going to use four real computers, two real switches and a Raspberry Pi with a SD card.

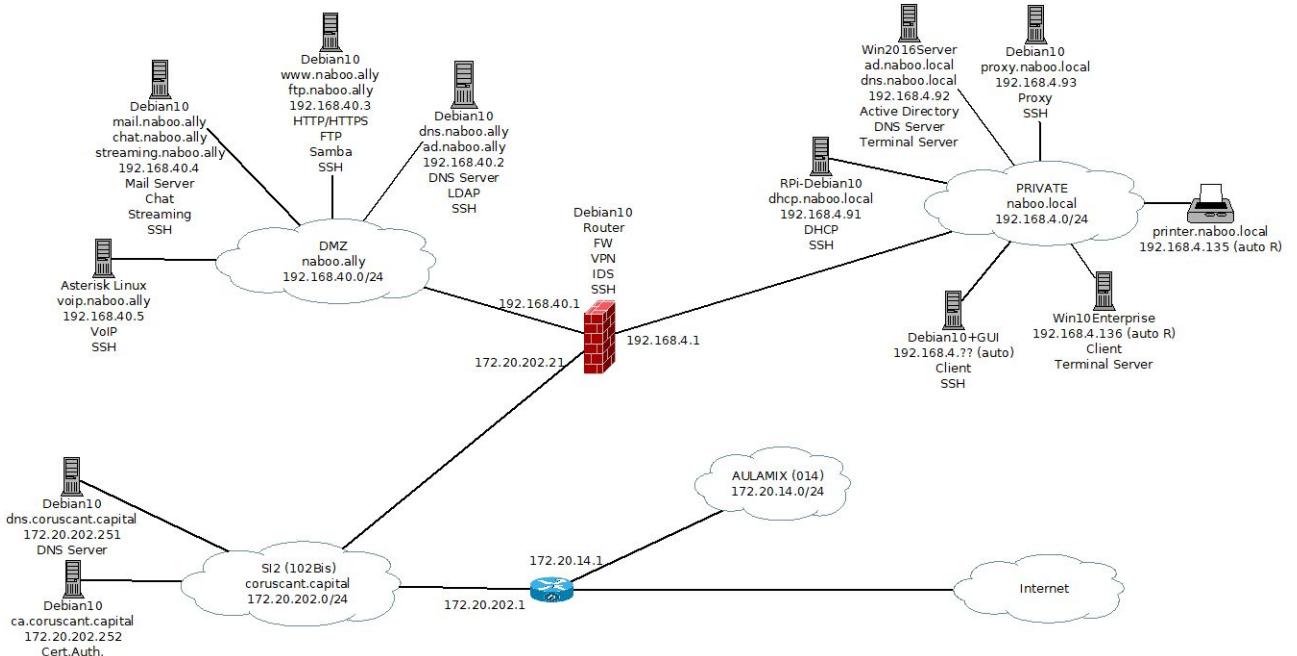
The goal is to create 2 new networks, being the third one the network located in the classroom which is going to give access to the internet to other networks. Those new networks are going to be used in the Naboo base, concretely located in the capital Theed.

The aim is create a solid and secure network with all the necessary services to defeat the Galactic Empire.

Naboo.local (192.168.4.0) is our private network in where the DHCP server (192.168.4.91), a printer (192.168.4.135), the Active Directory (Windows Server 2016, 192.168.4.92) and Proxy Server (Debian Linux, 192.168.4.93) are going to be placed.

Naboo.ally domain is the place where our public servers, the ones that can be accessible from everywhere, will be allocated. Here the Mail and Chat servers (Debian Linux, 192.168.40.4), the HTTP and FTP server (Debian Linux, 192.168.40.3) and LDAP server (Debian Linux, 192.168.40.2) are going to be placed.

We have planned the logical and the fisic schemas of this network, taking into account the correct distribution of all the necessary services and the purpose of maintaining the best security.



All the changes made in the Debian machines have been done with the “root” user.

In all the machines with Debian 10, is necessary to change the repository file because it won't be able to add any apt without changing it first.

So we open the /etc/apt/sources.list and by default there are 3 uncommented lines.

```
GNU nano 3.2                               /etc/apt/sources.list

#
# deb cdrom:[Debian GNU/Linux 10.1.0 _Buster_ - Official amd64 DVD Binary-1 20190908-01:09]/ buster$ deb cdrom:[Debian GNU/Linux 10.1.0 _Buster_ - Official amd64 DVD Binary-1 20190908-01:09]/ buster c$ deb http://security.debian.org/debian-security buster/updates main contrib deb-src http://security.debian.org/debian-security buster/updates main contrib
# buster-updates, previously known as 'volatile'
# A network mirror was not selected during install. The following entries
# are provided as examples, but you should amend them as appropriate
# for your mirror of choice.
#
# deb http://deb.debian.org/debian/ buster-updates main contrib
# deb-src http://deb.debian.org/debian/ buster-updates main contrib

[ Read 16 lines ]
^G Get Help   ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos  M-U Undo
^X Exit       ^R Read File   ^V Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line M-E Redo  Activ
```

We should comment the first line and add these lines:

Naboo
 Julen Fernández
 Ibon Pina
 SI2

```
GNU nano 3.2                               /etc/apt/sources.list

#
# deb cdrom:[Debian GNU/Linux 10.1.0 _Buster_ - Official amd64 DVD Binary-1 20190908-01:09]/ buster$ deb http://ftp.es.debian.org/debian buster main
deb-src http://ftp.es.debian.org/debian buster main
deb http://security.debian.org/debian-security buster/updates main contrib
deb-src http://security.debian.org/debian-security buster/updates main contrib
deb http://ftp.es.debian.org/debian buster-updates main contrib
deb-src http://ftp.es.debian.org/debian buster-updates main contrib

# buster-updates, previously known as 'volatile'
# A network mirror was not selected during install. The following entries
# are provided as examples, but you should amend them as appropriate
# for your mirror of choice.
#
# deb http://deb.debian.org/debian/ buster-updates main contrib
# deb-src http://deb.debian.org/debian/ buster-updates main contrib
```

When we have this file ready, we should install the ssh service in all the machines, in order to be able to control them from different workspaces.

All the virtual machines are going to be in bridge mode and the computers are going to be connected to a switch connected to the classroom network so that we have internet connection in all the machines.

To begin with the installation we had to check the state of the network cables.

There are going to be two main domains, one located in naboo.local and the other in naboo.ally.

Router

Installation of the network adapters

We have to add manually two networks adapters to the router so that we can configure the 3 networks. For this purpose, we open the computer and stick the network adapters to the computer.

OS Installation

For the installation of the debian router we need a debian iso. So we download the debian iso from the official website and use a pen drive in order to connect it to the router.

Once we have the usb with the iso connected to the PC, we boot it, and we have to access the BIOS in order to set the boot priority list. The only thing we have to do there is to choose the usb we are gonna use as the principal boot option.

After this, we save and apply the changes and we boot the pc again. The installation of debian should start right away. In the installation we just have to choose the non-gui option and install it.

Network Configuration of the 3 adapters

We have 3 network adapters, 2 of them connected via PCI and one implemented in the motherboard. The first step is to get administrator privileges, we can do this using the command “su -” and then writing the password.

Once we have administrator privileges we have to access the file where the configuration of the network interfaces is done: /etc/network/interfaces.

Inside this file have set these configurations for each interface:

```

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
iface lo inet loopback

iface enp2s1 inet static
    address 172.20.202.21
    netmask 255.255.255.0
    gateway 172.20.202.1
    dns-nameservers 8.8.8.8

iface enp2s5 inet static
    address 192.168.4.1
    netmask 255.255.255.0
    gateway 172.20.202.1

iface enp2s2 inet static
    address 192.168.40.1
    netmask 255.255.255.0

```

After doing that we have to restart the networking service so that it updates the interfaces information and it works properly.

The interface enp2s1 is going to be directly connected to the classroom network (172.20.202.0).

The interface enp2s5 is going to be connected to switch B(192.166.4.0).

The interface enp2s2 is going to be connected to switch A(192.166.40.0).

Allow internet access

In order to allow the internet access through our router we have to enter some commands:

First we have to activate routing. We just have to set a “1” in this file: /proc/sys/net/ipv4/ip_forward.

But to do this change permanent we have to edit the file /etc/sysctl.conf and change again the “0” to a “1”.

The second step is to masquerade the output of our network to the internet, we can do this using this command:

```
iptables -t nat -A POSTROUTING -o enp2s1 -j MASQUERADE
```

Startup script

Naboo
 Julen Fernández
 Ibon Pina
 SI2

The main step of doing the script is just to take the necessary commands that we used in the step above and put them in a script in the /etc/init.d folder.

The final script looks something like this:

```
GNU nano 3.2

#!/bin/bash

### BEGIN INIT INFO
# Provides: fw.sh
# Required-start: $all
# Required-stop: $all
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: Firewall
# Description: Establece el firewall en este router
### END INIT INFO

#Activar enrutamiento
echo 1 > /proc/sys/net/ipv4/ip_forward

#Enmascarar la salida a internet
iptables -A FORWARD -j ACCEPT
iptables -t nat -A POSTROUTING -o enp2s1 -j MASQUERADE
```

But this won't work at the startup of the machine if you don't use this command:

```
root@debian-router:/etc/init.d# update-rc.d script.sh defaults
root@debian-router:/etc/init.d# ls -l /etc/rc5.d/
total 0
lrwxrwxrwx 1 root root 26 Sep 23 09:48 S01console-setup.sh -> ../init.d/console-setup.sh
lrwxrwxrwx 1 root root 14 Sep 23 09:46 S01cron -> ../init.d/cron
lrwxrwxrwx 1 root root 14 Sep 23 09:50 S01dbus -> ../init.d/dbus
lrwxrwxrwx 1 root root 17 Sep 23 09:46 S01rsyslog -> ../init.d/rsyslog
lrwxrwxrwx 1 root root 19 Sep 26 09:02 S01script.sh -> ../init.d/script.sh
lrwxrwxrwx 1 root root 13 Sep 24 10:22 S01ssh -> ../init.d/ssh
root@debian-router:/etc/init.d#
```

As we have been progressing in the challenge we have added some more lines to the script:

```
#Eliminar reglas actuales
iptables -F
iptables -t nat -F

#Politica por defecto ACEPTAR
iptables -P INPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -P OUTPUT ACCEPT
```

The first 2 commands are just to remove the actual iptables rules, and the other 3 are to accept anything that comes into the firewall.

In order to connect via ssh from the Aulamix class (network 172.20.14.0/24) and from our class network we had to add some more lines in the script so that we could redirect each connection using a different port to each machine.

```
#Para acceder desde Aulamix
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.14.0/24 -p tcp --dport 2493 -j DNAT --to 192.168.4.93:22
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.14.0/24 -p tcp --dport 2484 -j DNAT --to 192.168.40.4:22
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.14.0/24 -p tcp --dport 2402 -j DNAT --to 192.168.40.2:22
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.14.0/24 -p tcp --dport 2483 -j DNAT --to 192.168.40.3:22
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.14.0/24 -p tcp --dport 2405 -j DNAT --to 192.168.40.5:22
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.14.0/24 -p tcp --dport 2042 -j DNAT --to 192.168.4.2:22
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.14.0/24 -p tcp --dport 2491 -j DNAT --to 192.168.4.91:22

#Para acceder desde Si2
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.202.0/24 -p tcp --dport 2493 -j DNAT --to 192.168.4.93:22
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.202.0/24 -p tcp --dport 2404 -j DNAT --to 192.168.40.4:22
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.202.0/24 -p tcp --dport 2402 -j DNAT --to 192.168.40.2:22
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.202.0/24 -p tcp --dport 2403 -j DNAT --to 192.168.40.3:22
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.202.0/24 -p tcp --dport 2405 -j DNAT --to 192.168.40.5:22
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.202.0/24 -p tcp --dport 2042 -j DNAT --to 192.168.4.2:22
iptables -t nat -A PREROUTING -i enp2s1 -s 172.20.202.0/24 -p tcp --dport 2491 -j DNAT --to 192.168.4.91:22
```

SSH Asymmetric cryptography:

In order to access remotely and in a secure way to our machines, we are going to use ssh, ssh is a network protocol that gives users, particularly system administrators, a secure way to access a computer over an unsecured network.

We are going to install the “openssh-server” package in order to install both client and server ssh.

The first step is to have ssh installed in both the server and the client used to connect via ssh.

Once we have installed the ssh server these are the steps we are gonna follow:

1 - Use the command “ssh-keygen” to generate a new pair of keys in our client and set a passphrase while doing it

```
debian-10-2@dns:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/debian-10-2/.ssh/id_rsa):
Created directory '/home/debian-10-2/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/debian-10-2/.ssh/id_rsa.
Your public key has been saved in /home/debian-10-2/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:aphM81X/F5s7o5oEfks0lUDZf8HaBy9nyZGK957oan8 debian-10-2@dns.naboo.ally
The key's randomart image is:
+---[RSA 2048]----+
 .oo   .
 |....+.o|
 |.+.oo++o|
 |.*.o.++=|
 . . . S + ..o=
 = + o o o . =
 = o . + . =
 . + o Eo.
 ..=oo..o |
+---[SHA256]----+
debian-10-2@dns:~$
```

2 - You can use the command “ssh-keygen -p to change the passphrase if needed.

```
debian-10-2@dns:~$ ssh-keygen -p
Enter file in which the key is (/home/debian-10-2/.ssh/id_rsa):
Enter old passphrase:
Key has comment 'debian-10-2@dns.naboo.ally'
Enter new passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved with the new passphrase.
debian-10-2@dns:~$
```

3 - With the command “ssh-copy-id -i .ssh/id_rsa.pub -p 2021

debian-router@192.168.4.1” you will be able to copy the public key of your client to the “authorized_keys” file of the user in the router.

```
debian-10-2@dns:~$ ssh-copy-id -i .ssh/id_rsa.pub -p 2021 debian-router@192.168.40.1
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: ".ssh/id_rsa.pub"
The authenticity of host '[192.168.40.1]:2021 ([192.168.40.1]:2021)' can't be established.
ECDSA key fingerprint is SHA256:4du2atnR8223pwUfd1mgfHSGiKvZjlmhu7czVjk14Tc.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
debian-router@192.168.40.1's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh -p '2021' 'debian-router@192.168.40.1'"
and check to make sure that only the key(s) you wanted were added.

debian-10-2@dns:~$
```

4 - Once the key is copied into the server you have to add lines in the config file of the ssh in the server, to do this you will have to edit it using the command “nano /ssh/sshd_config” while being logged with the user of the router and not with the “root” user.

Then you add this 3 lines and also change the “PasswordAuthentication yes” to “no” so that ssh asks for the passphrase on every login.

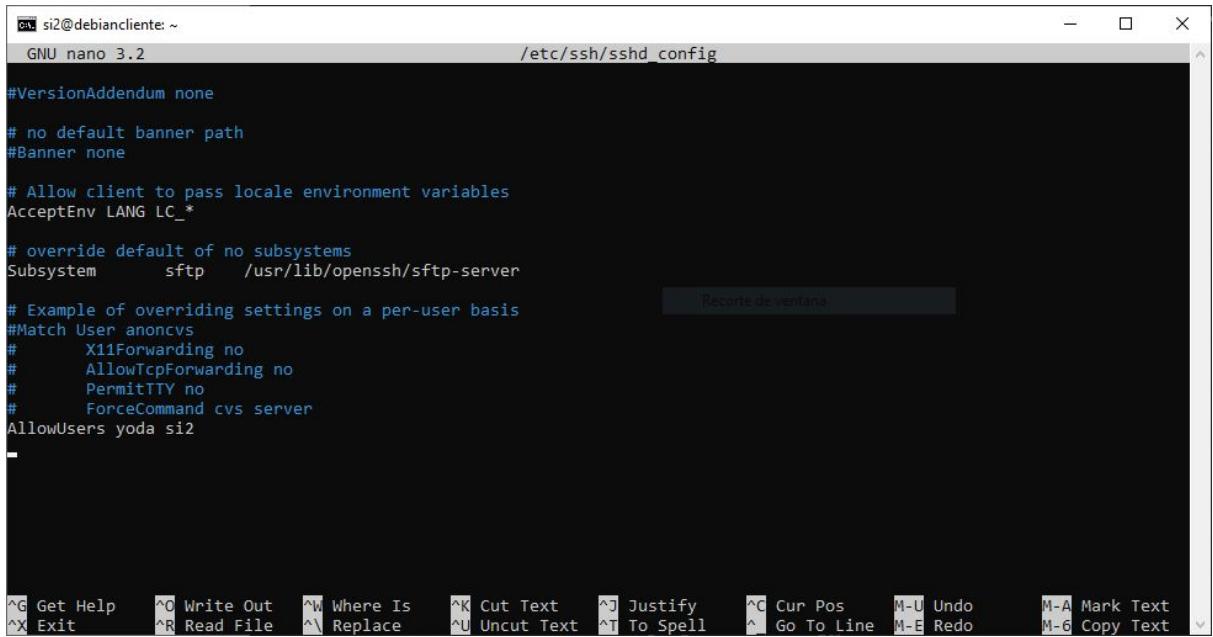
```
RSAAuthentication yes
PubkeyAuthentication yes
AuthorizedKeysFile  %h/.ssh/authorized_keys
```

```
debian-10-2@dns:~$ ssh -p 2021 debian-router@192.168.40.1
Enter passphrase for key '/home/debian-10-2/.ssh/id_rsa':
Linux Router 4.19.0-6-amd64 #1 SMP Debian 4.19.67-2 (2019-08-28) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Oct 11 06:33:26 2019 from 192.168.40.133
debian-router@Router:~$
```

Login with ssh against the active directory.



The screenshot shows a terminal window titled "si2@debiancliente: ~" running the "nano 3.2" editor. The file being edited is "/etc/ssh/sshd_config". The configuration file contains several lines of text, including comments and specific directives for the SSH server. At the bottom of the screen, there is a menu bar with various keyboard shortcuts for navigating and editing the text.

```
si2@debiancliente: ~
GNU nano 3.2
/etc/ssh/sshd_config

#VersionAddendum none
# no default banner path
#Banner none

# Allow client to pass locale environment variables
AcceptEnv LANG LC_*
# override default of no subsystems
Subsystem      sftp    /usr/lib/openssh/sftp-server
# Example of overriding settings on a per-user basis
#Match User anoncvs
#       X11Forwarding no
#       AllowTcpForwarding no
#       PermitTTY no
#       ForceCommand cvs server
AllowUsers yoda si2

^G Get Help      ^O Write Out   ^W Where Is     ^K Cut Text    ^J Justify    ^C Cur Pos     M-U Undo
^X Exit          ^R Read File   ^\ Replace      ^U Uncut Text  ^T To Spell   ^A Go To Line  M-E Redo
                                         M-A Mark Text  M-6 Copy Text
```

Webgraphy:

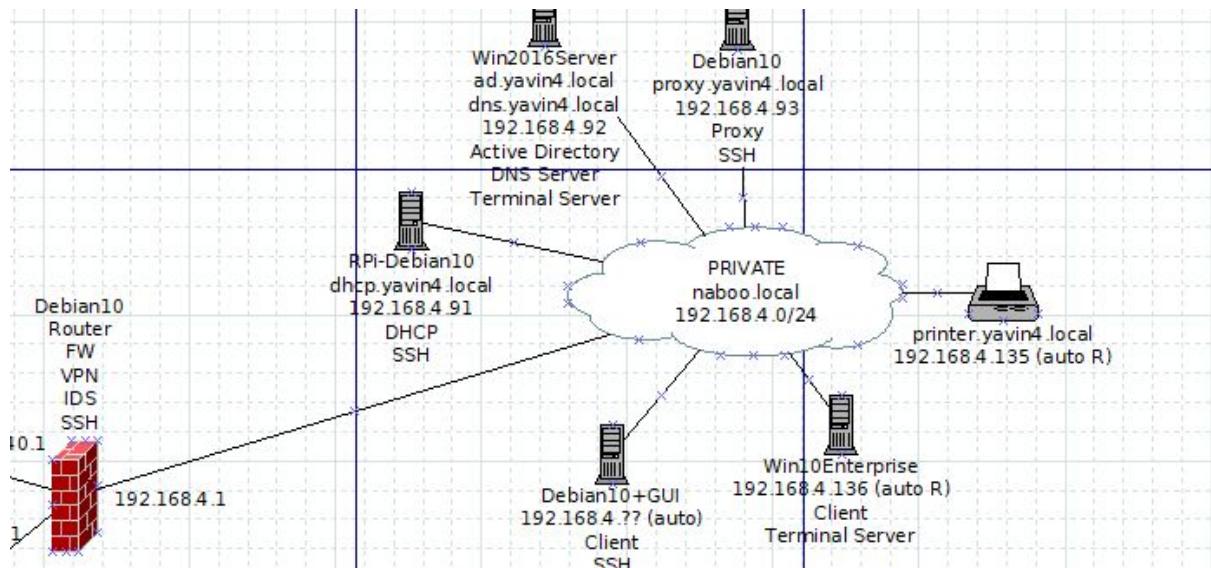
<https://stackoverflow.com/questions/6377009/adding-public-key-to-ssh-authorized-keys-does-not-log-me-in-automatically>

<https://www.digitalocean.com/community/questions/error-permission-denied-publickey-when-i-try-to-ssh>

<https://www.ssh.com/ssh/keygen/>

http://www.I3jane.net/doc/linux/suse/suselinux-adminguide_es/ch12s04.html

Complete configuration of naboo.local

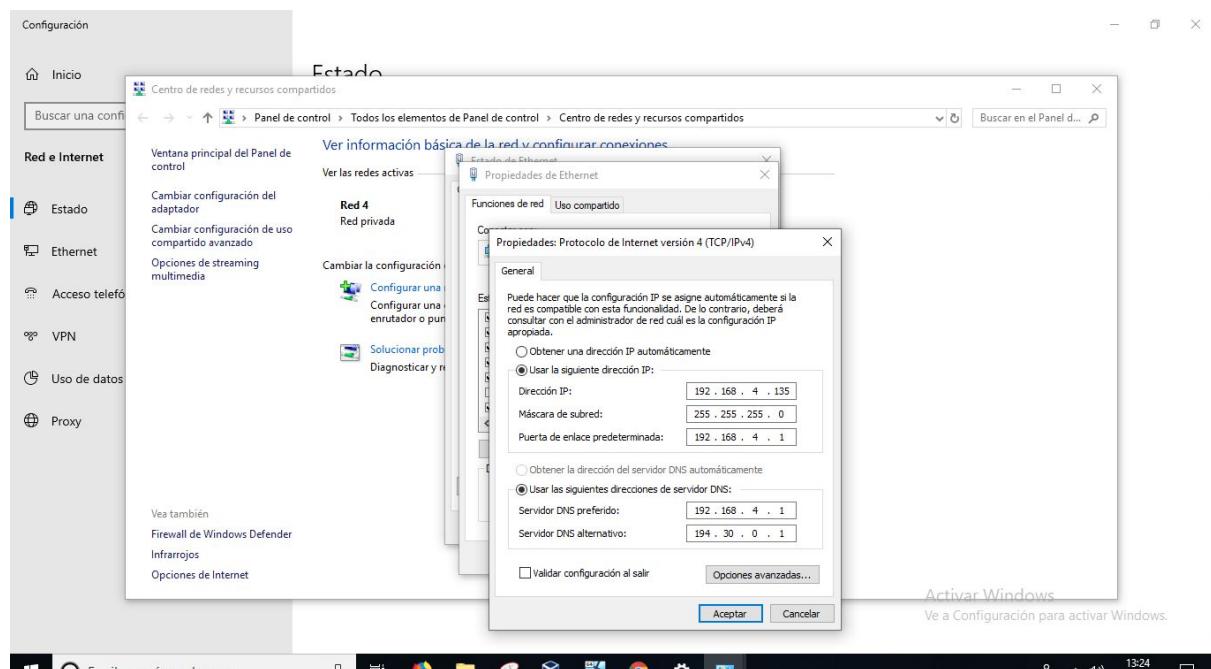


Domain and Naboo.local DNS

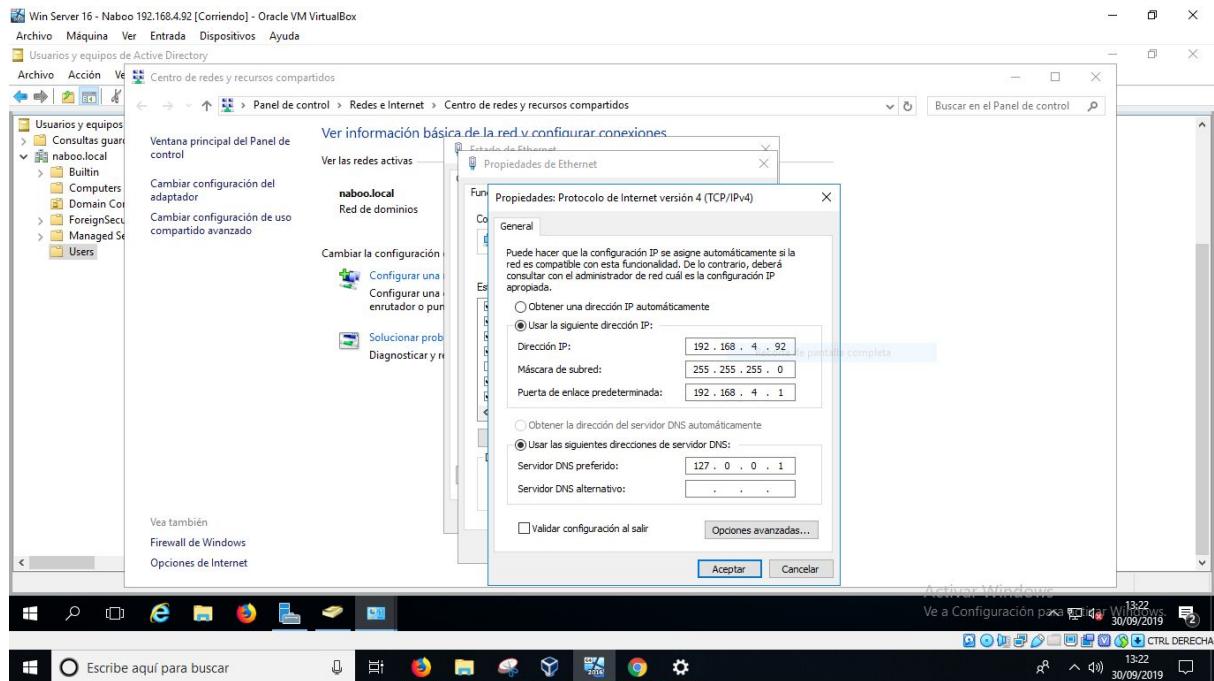
Once we have all the machines from Virtual Box ready, we have to change the network adapter to bridge mode so that our Windows Server 2016 can acces the internet as it is connected to SI2-05 PC(192.168.4.135). This computer is also connected to the switch B (192.168.4.0 network)

Network Configuration of WS2016

The PC SI2-05 will be connected to the switch with this IP.



Después le cambiamos la IP al Windows Server 2016 y de esta manera tendrá acceso a internet.

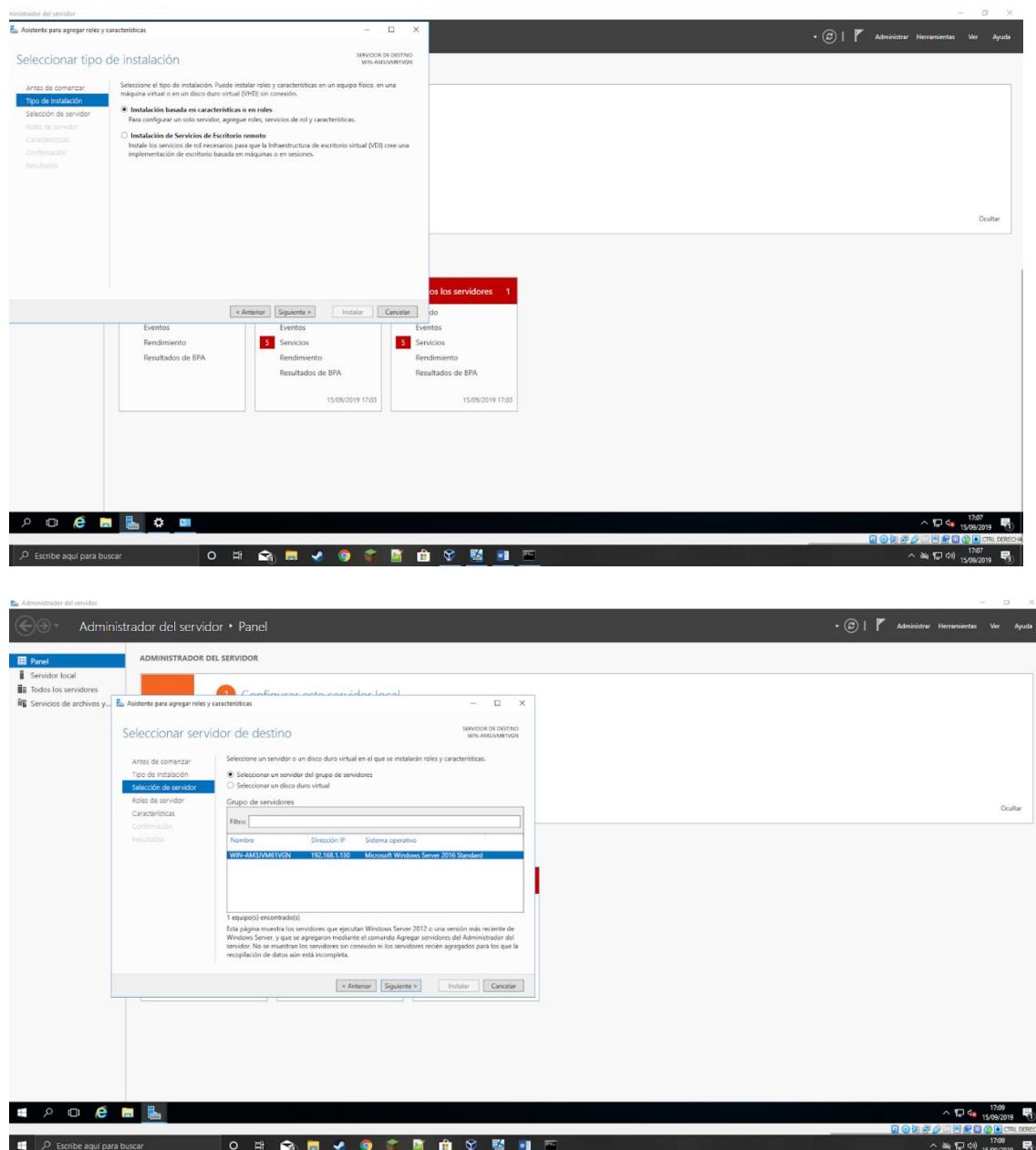


Active Directory:

Para poder crear el dominio, tendremos que instalar el Servicio de Directorio Activo, para ello en el Administrador del Servidor.

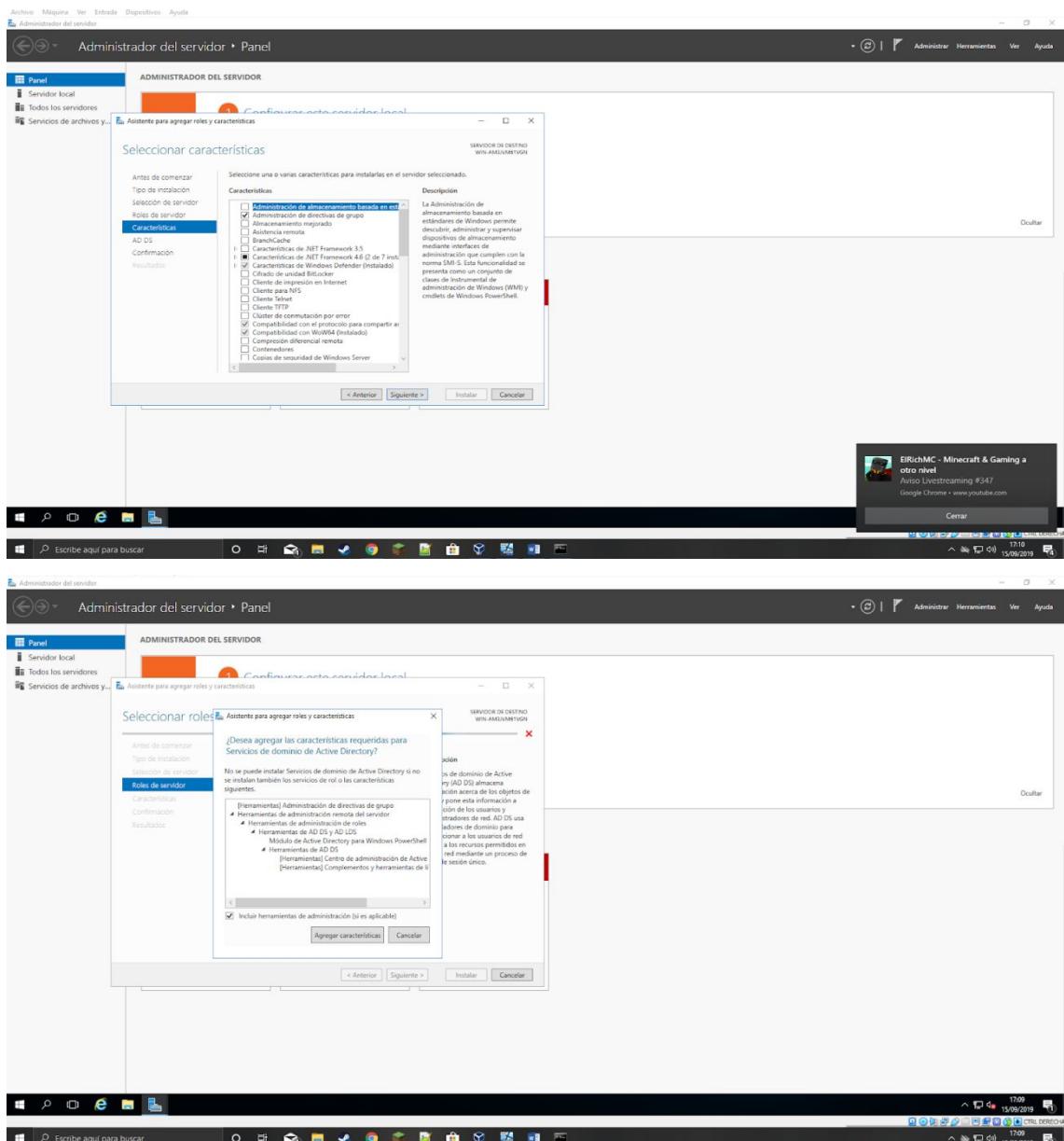
We will need to install the role of the Directory Active Service from the Server Administrator tool.

We should click in manage and add a new role.

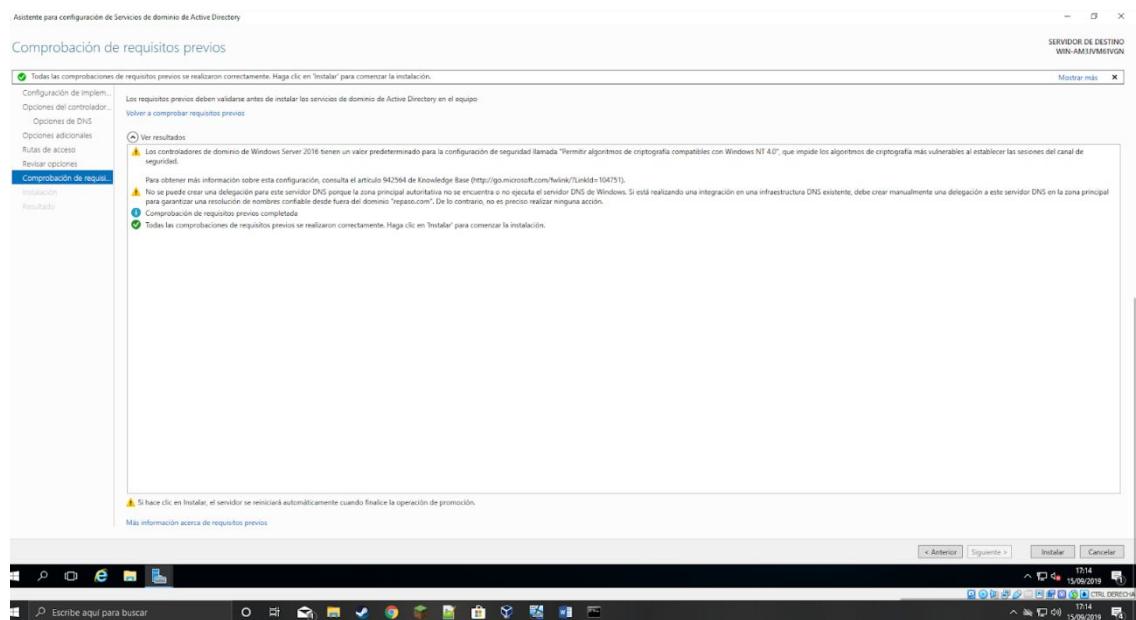
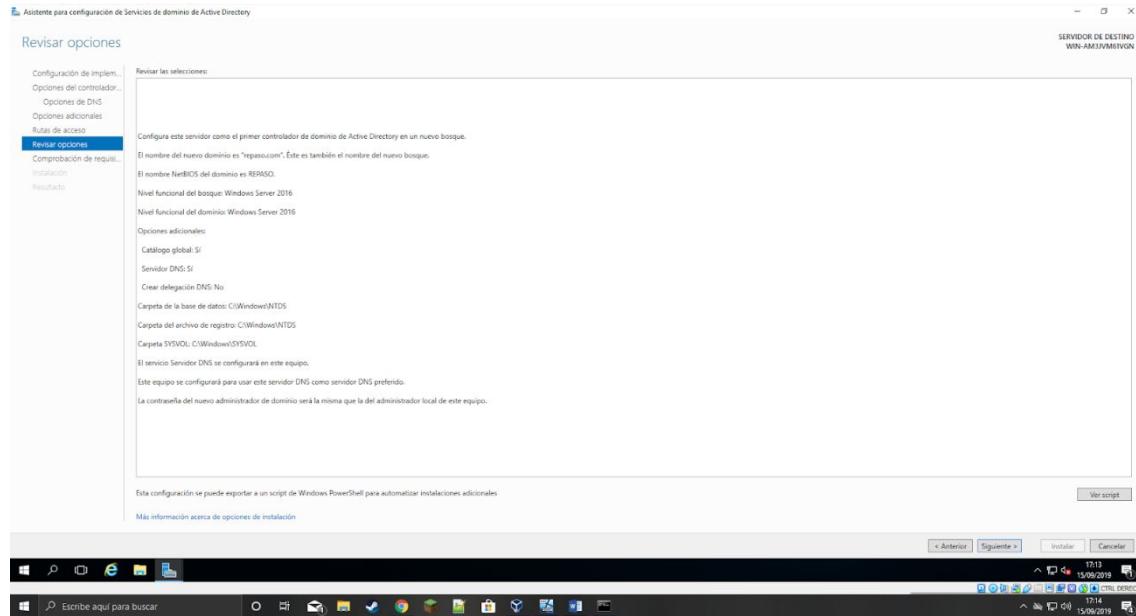


We have to search the Active Directory Domain services and complete the instalation.

Naboo
 Julen Fernández
 Ibon Pina
 SI2

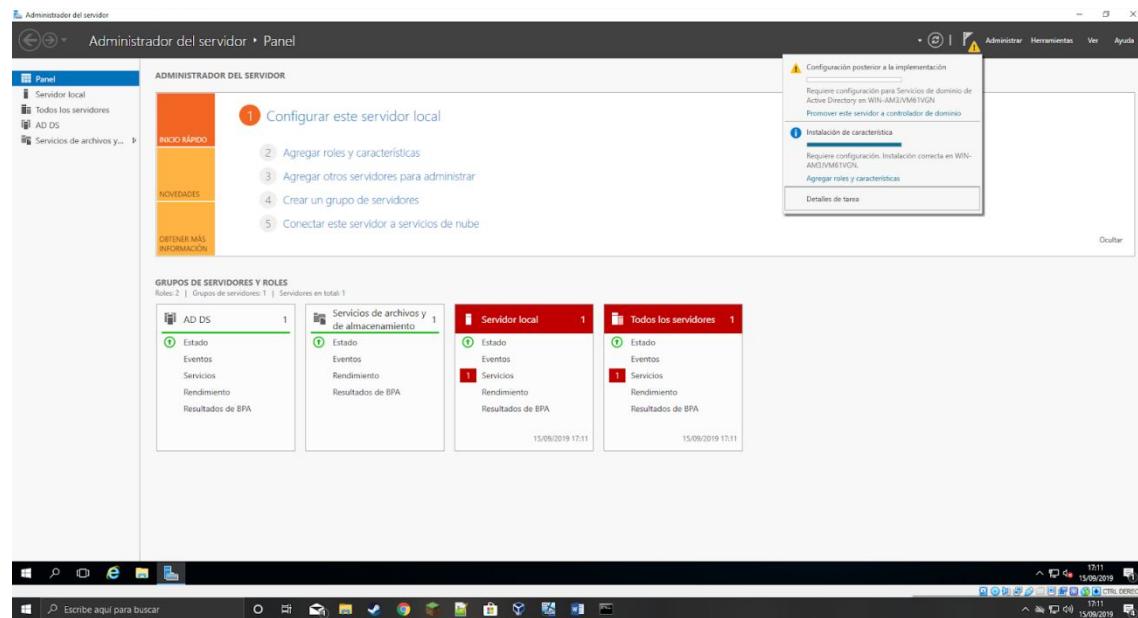


Naboo
 Julen Fernández
 Ibon Pina
 SI2



Naboo
Julen Fernández
Ibon Pina
SI2

Dominio: naboo.local

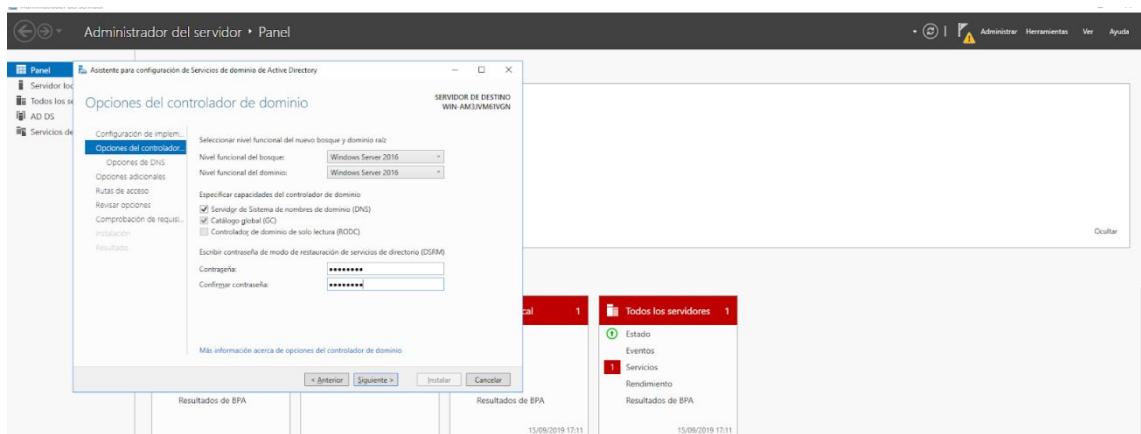
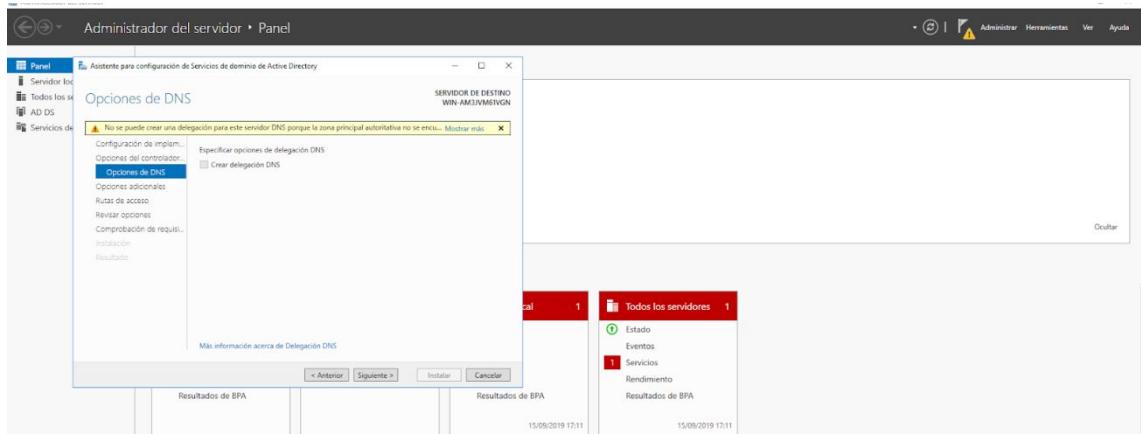


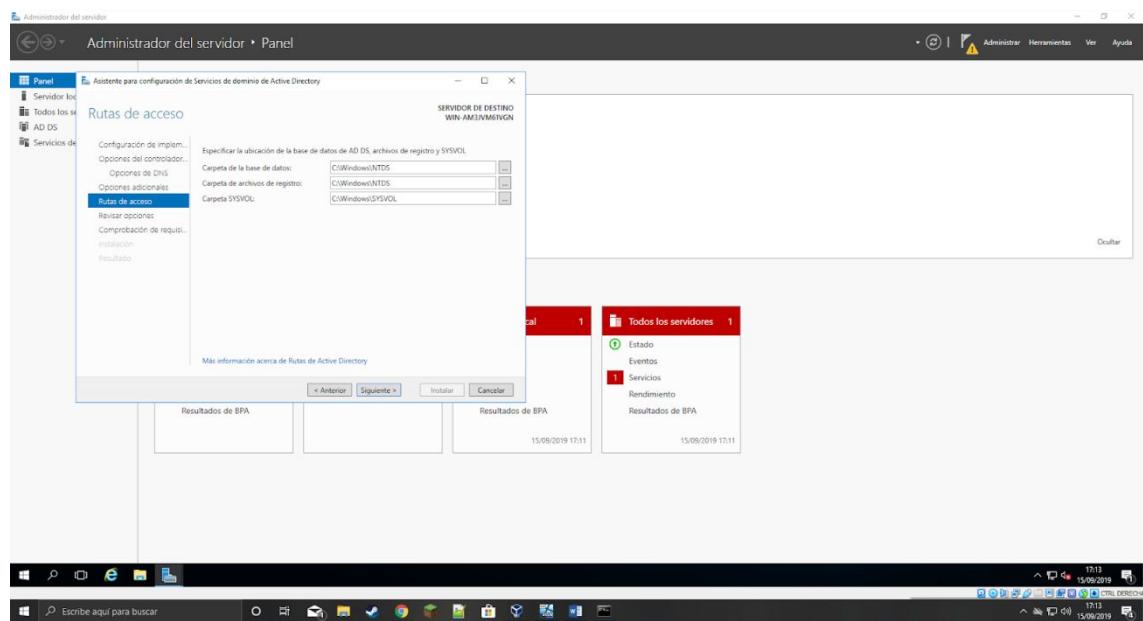
Once we reboot the machine, we will have a notification in the Server Administrator tool. We should click on it and promote the server to a domain controller.

Una vez dado, nos abre una ventada donde ya empezaremos a crear nuestro dominio.

Seleccionamos la Opción de “Agregar un nuevo bosque” y abajo ponemos el nombre de nuestro dominio, en nuestro caso “naboo.local”

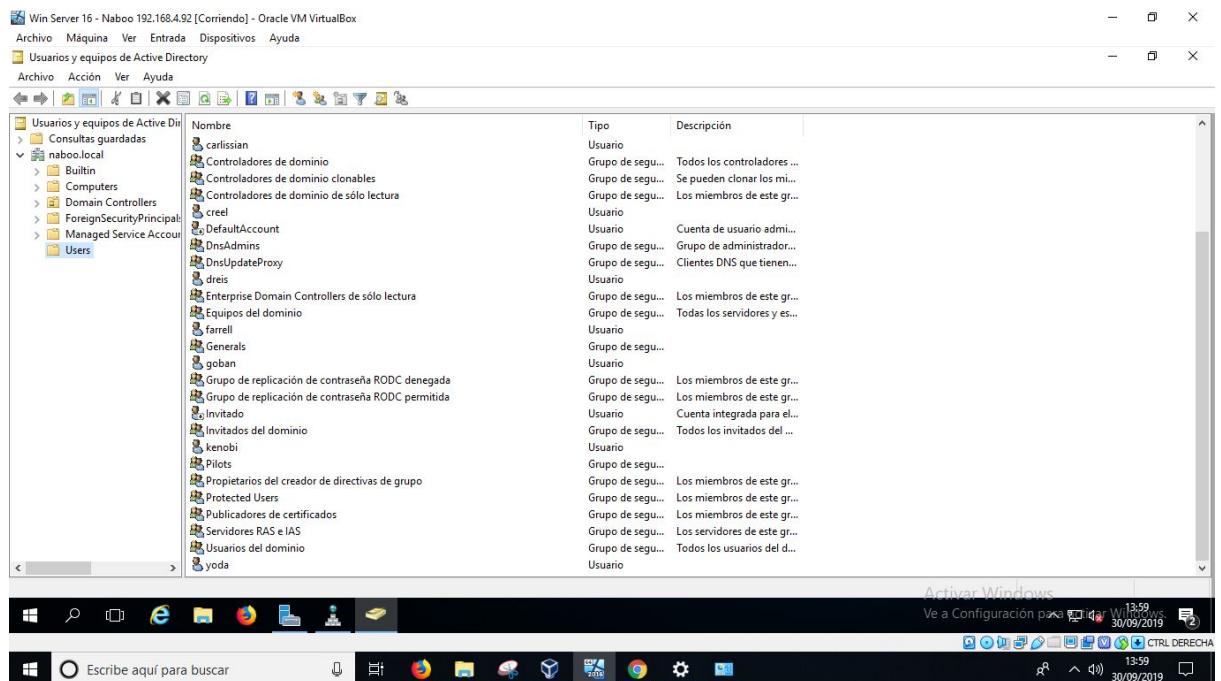
Las otras dos opciones son válidas para cuando ya hay un dominio creado, una para crear un controlador de dominio y la otra para crear otro dominio.





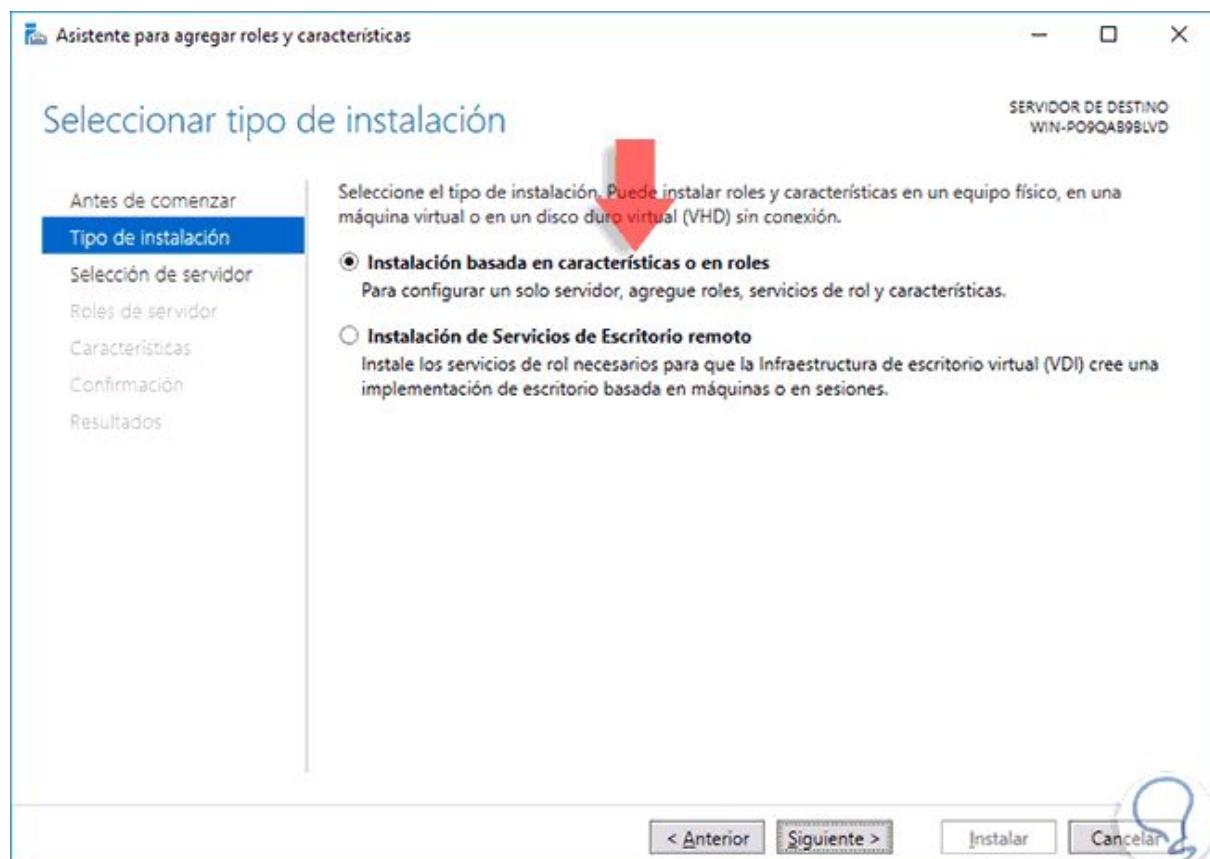
Creación Usuarios y Grupos

Nombre	Tipo	Descripción
Administrator	Usuario	Cuenta integrada para la administración.
Administradores clave	Grupo de seguimiento	Los miembros de este grupo tienen acceso a las características de administración avanzadas.
Administradores clave de la organización	Grupo de seguimiento	Los miembros de este grupo tienen acceso a las características de administración avanzadas.
Administradores de empresas	Grupo de seguimiento	Administradores designados.
Administradores de esquema	Grupo de seguimiento	Administradores designados.
Administradores del dominio	Grupo de seguimiento	Administradores designados.
bridge	Usuario	
Captains	Grupo de seguimiento	
carlodian	Usuario	
Controladores de dominio	Grupo de seguimiento	Todos los controladores de dominio.
Controladores de dominio clonables	Grupo de seguimiento	Se pueden clonar los miembros.
Controladores de dominio de sólo lectura	Grupo de seguimiento	Los miembros de este grupo no pueden modificar los datos.
creel	Usuario	
DefaultAccount	Usuario	Cuenta de usuario administrador.
DnsAdmins	Grupo de seguimiento	Grupos de administrador de DNS.
DnsUpdateProxy	Grupo de seguimiento	Cuentas que tienen permiso para actualizar el directorio.
dries	Usuario	
Enterprise Domain Controllers de sólo lectura	Grupo de seguimiento	Los miembros de este grupo no pueden modificar los datos.
Equipos del dominio	Grupo de seguimiento	Todas las servidores y equipos.
farrell	Usuario	
Generals	Grupo de seguimiento	
goban	Usuario	
Grupo de replicación	Grupo de seguimiento	Los miembros de este grupo replican datos.
Grupo de replicación	Grupo de seguimiento	Los miembros de este grupo replican datos.
Invitado	Usuario	Cuenta integrada para el invitado.
Invitados del dominio	Grupo de seguimiento	Todos los invitados del dominio.



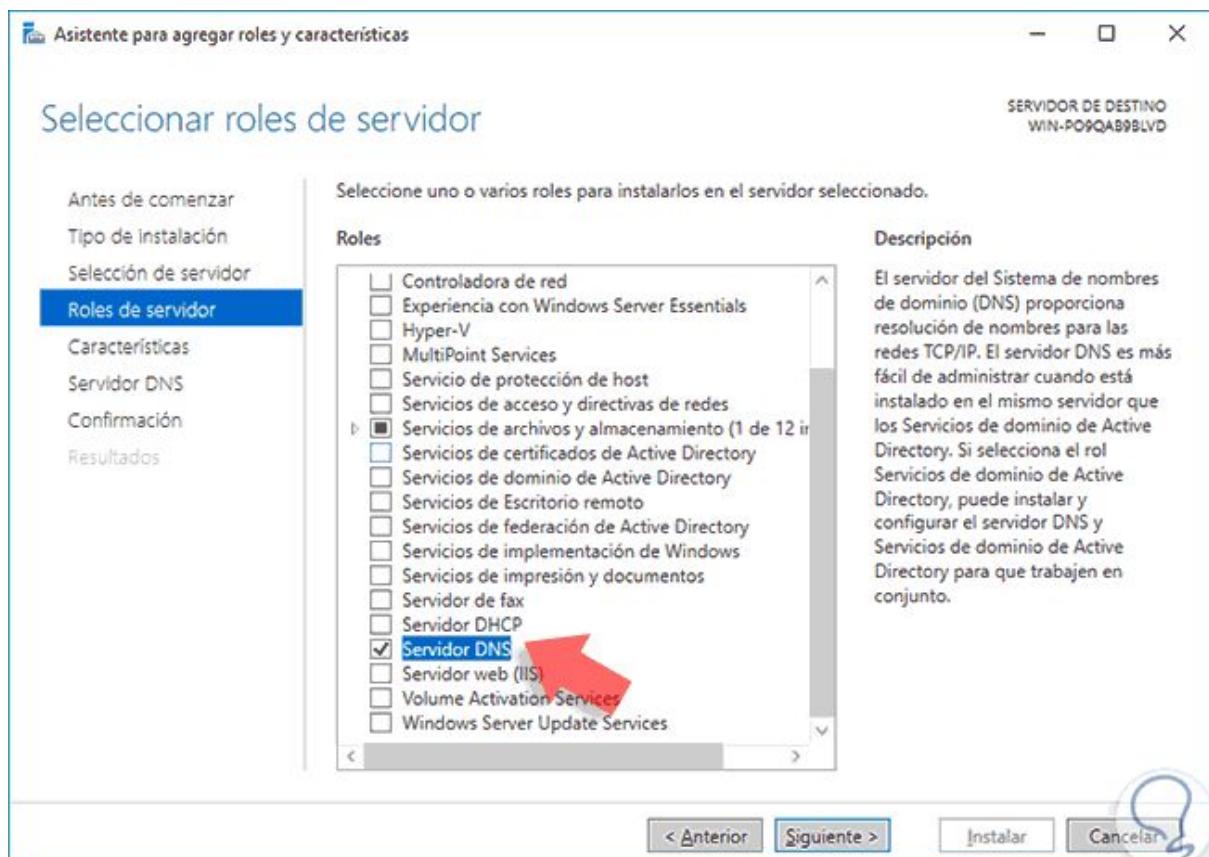
DNS

Para instalar el DNS lo primero que hay que hacer es ir a agregar roles y características

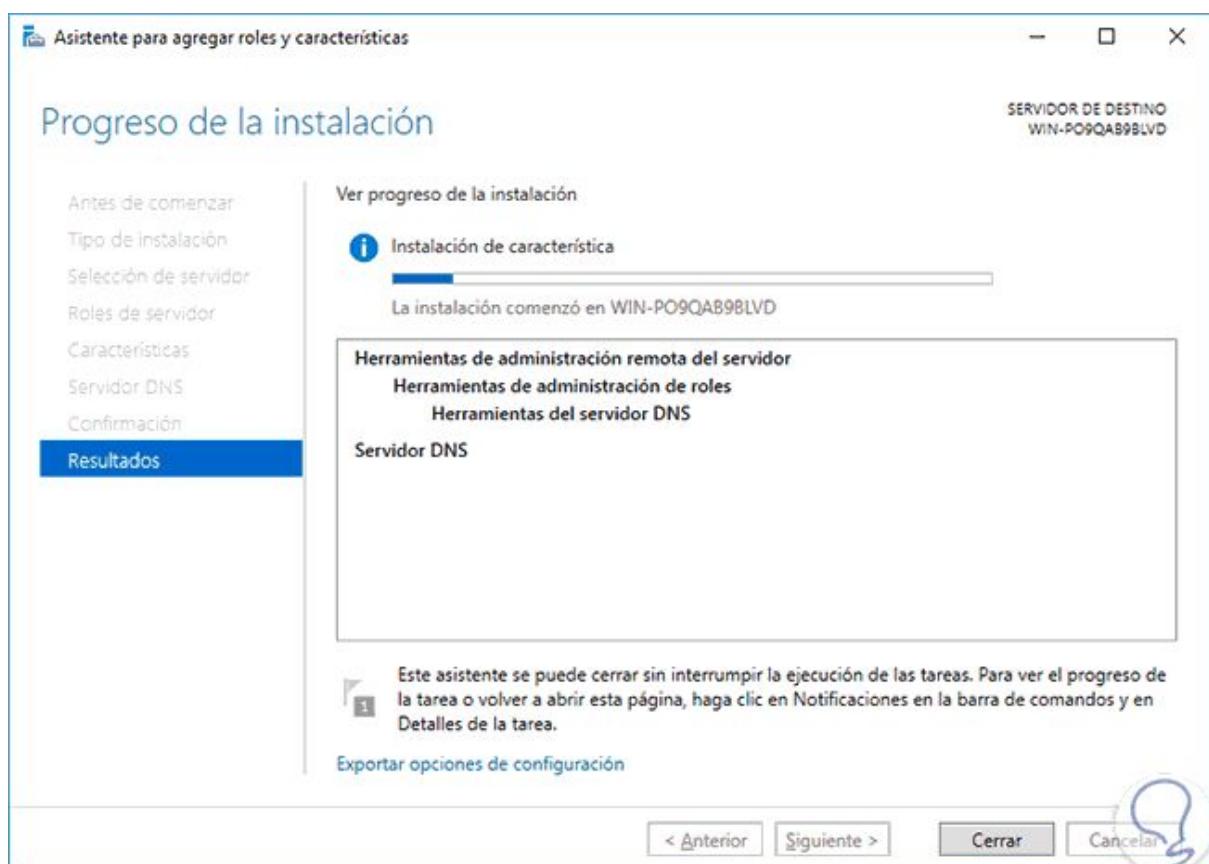


Elegimos el servidor dns y le damos a siguiente

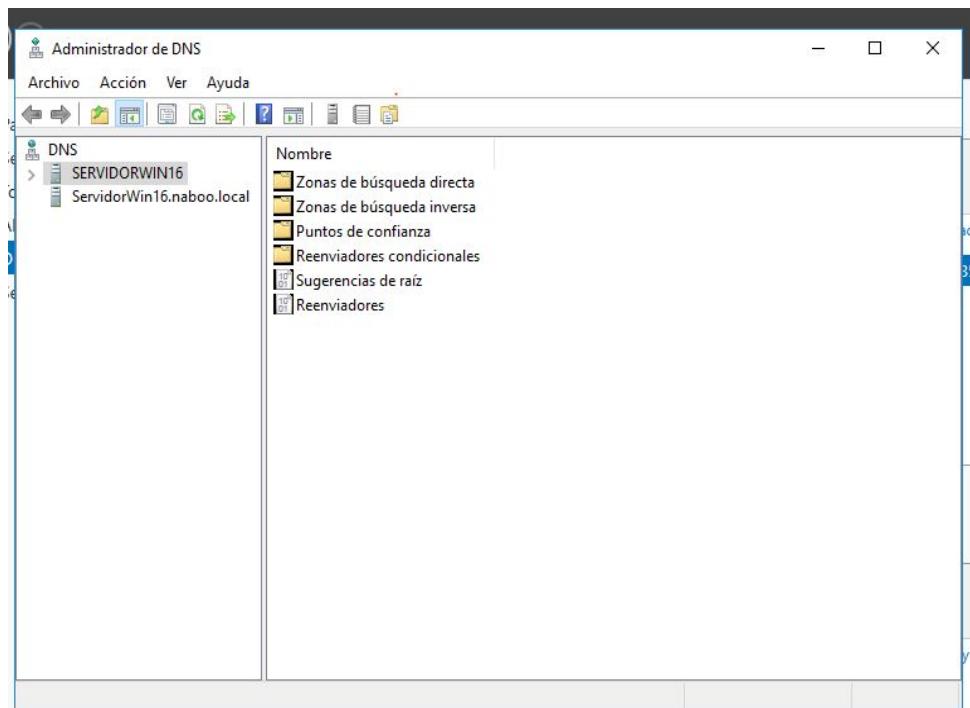
Naboo
Julen Fernández
Ibon Pina
SI2



Por último le damos a instalar



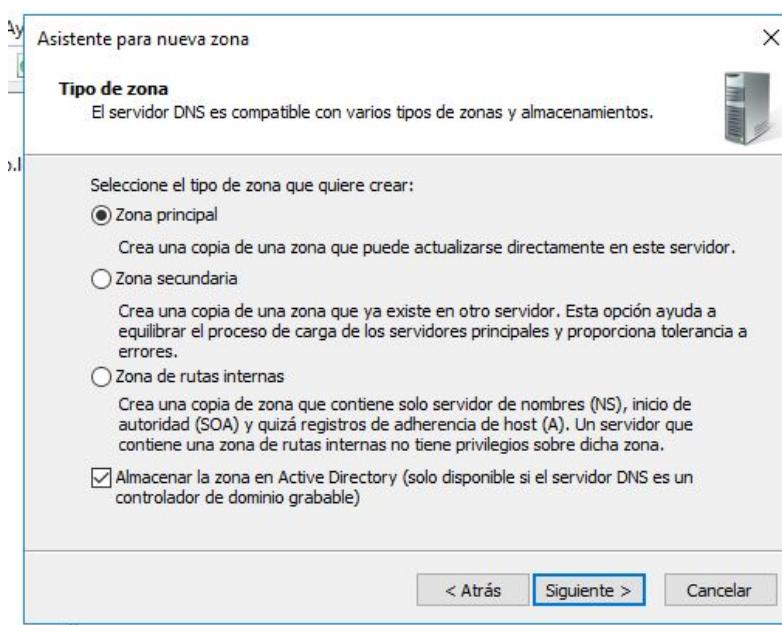
El siguiente paso es abrir el administrador del dns, que se encuentra en la parte superior derecha del administrador del servidor, en herramientas.



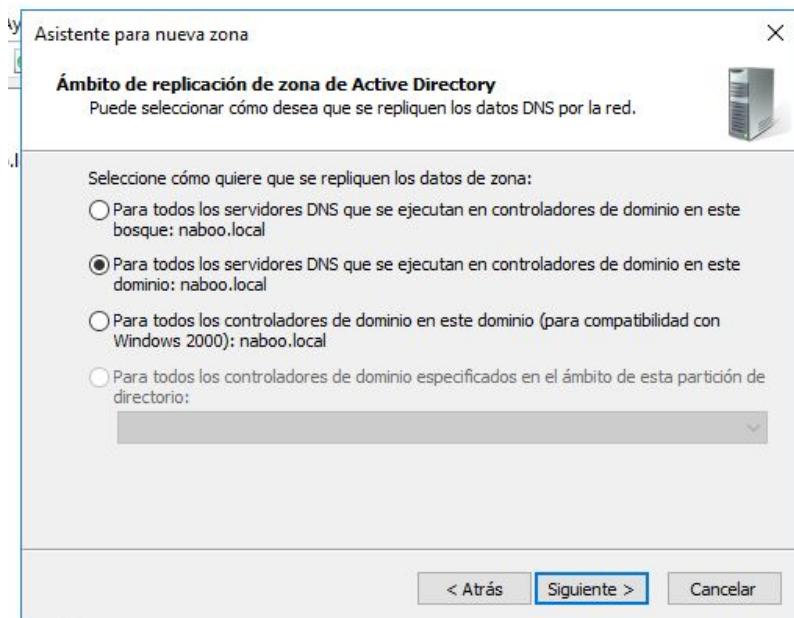
Ahora toca configurar el dns:

Le damos click derecho a nuestro servidor y crear zona nueva

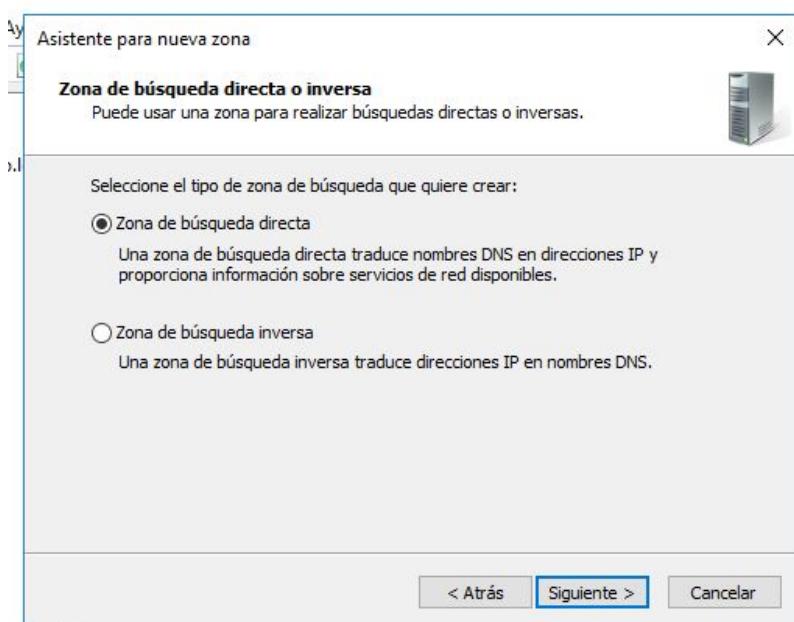
Escogemos Zona primaria



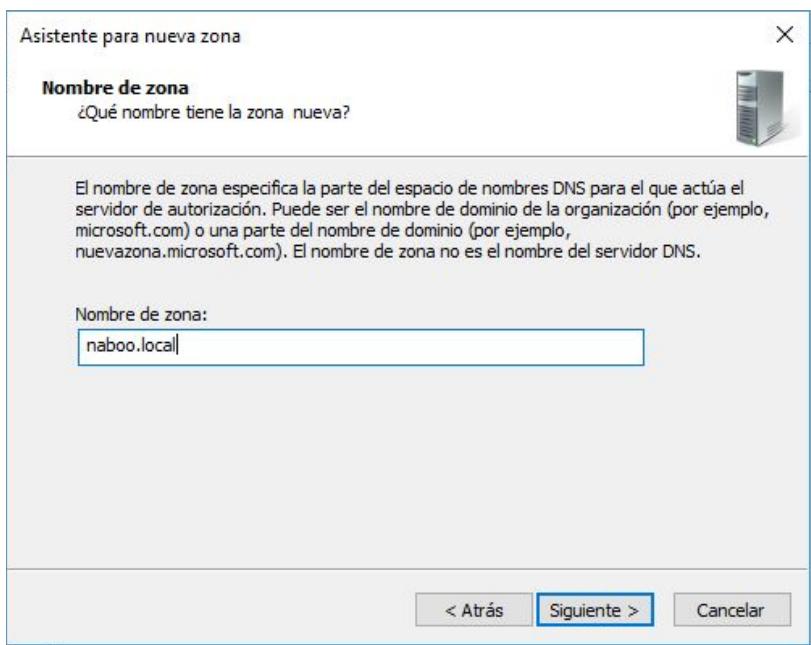
Aquí dejamos la opción por defecto:



De momento la zona que queremos crear es directa, la inversa la crearemos más tarde:



Le ponemos un nombre, en nuestro caso será naboo.local



Por último le damos a finalizar



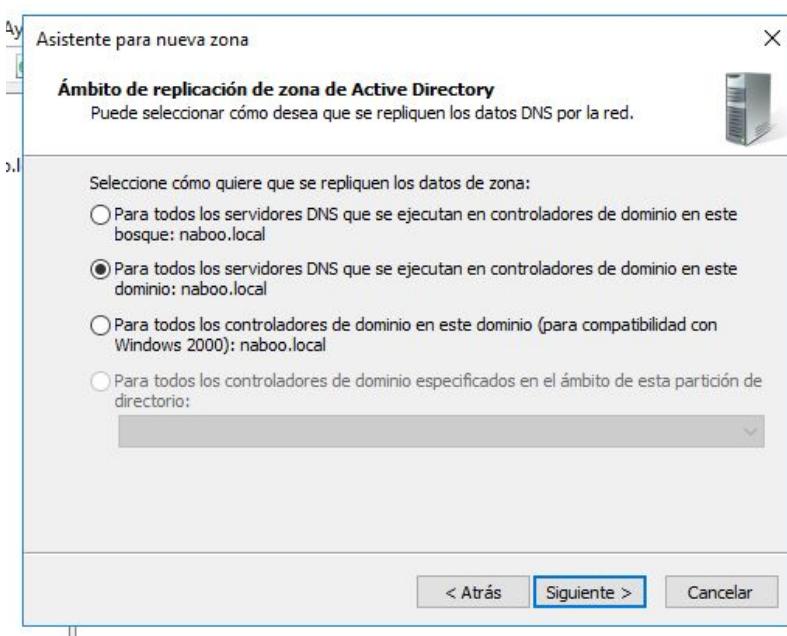
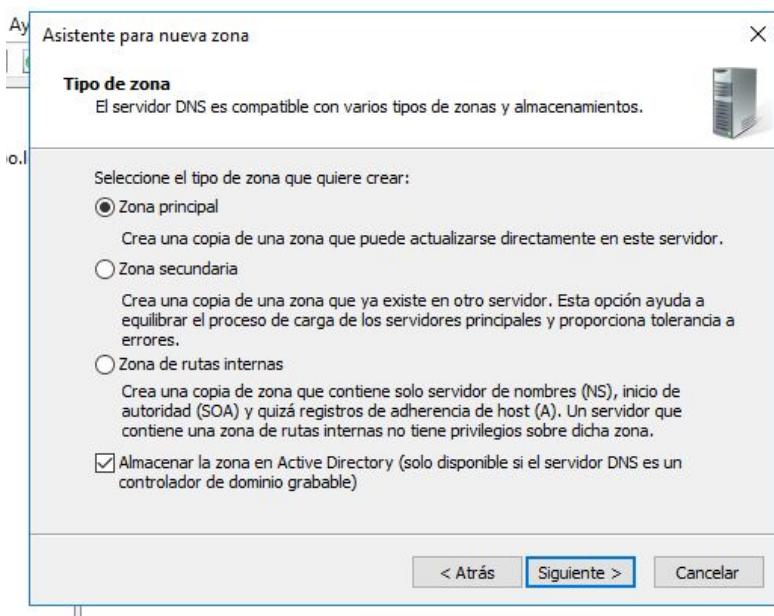
Lo primero es añadir los hosts de nuestra red local (192.168.4.0/24) para poder identificar quien es quien desde los equipos.

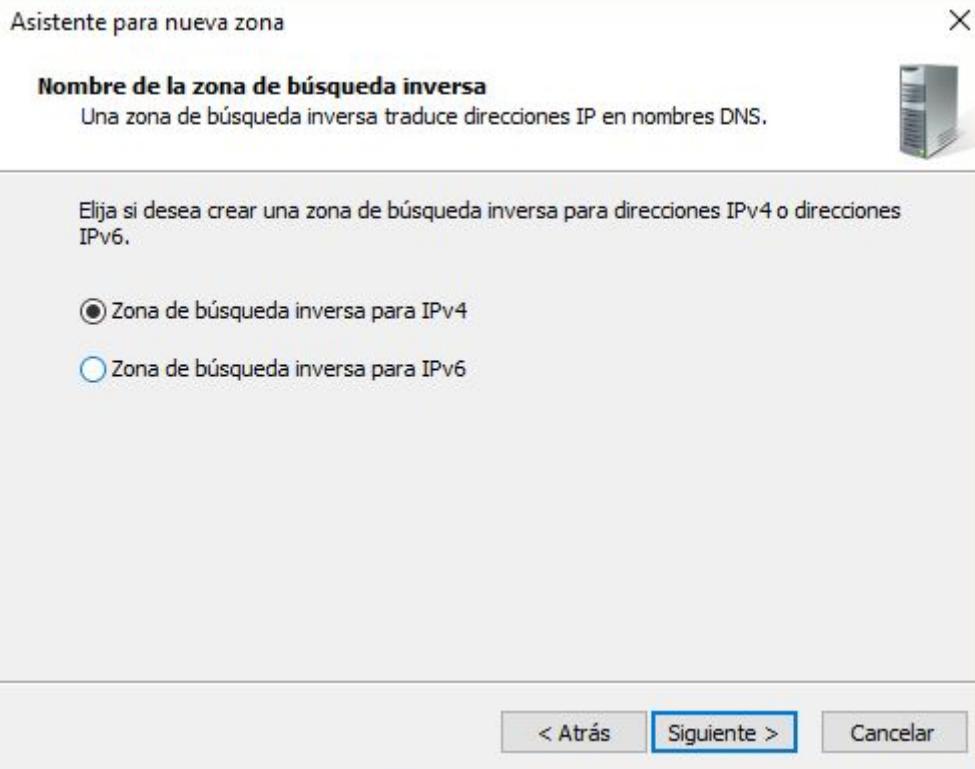
Para esto debemos entrar en la zona que hemos creado y una vez allí creamos cada host de la red y le especificamos su ip y su nombre.

Por ejemplo dhcp.naboo.local 192.168.4.91

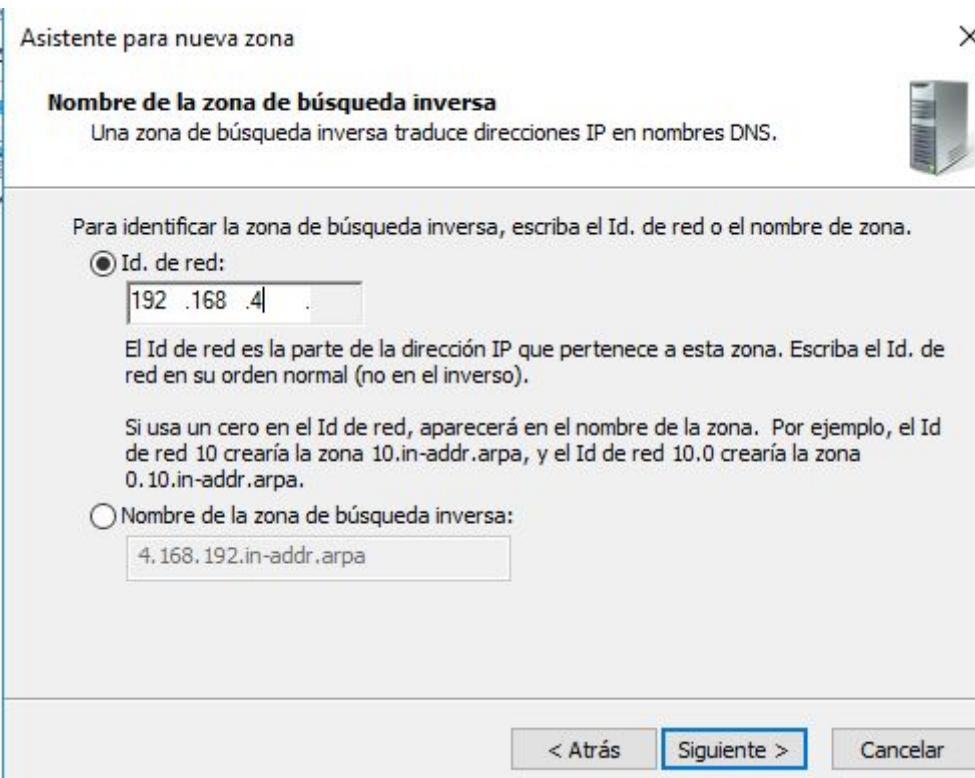
ad	Host (A)	192.168.4.92	static
debianclient	Host (A)	192.168.4.2	static
dhcp	Host (A)	192.168.4.91	static
proxy	Host (A)	192.168.4.93	static
servidorwin16	Host (A)	192.168.4.92	static
winclient	Host (A)	192.168.4.136	static

La zona directa ya está creada, ahora solo falta crear la zona inversa, para ello daremos clic derecho en el apartado de zona de búsqueda inversa y crearemos la nueva zona, el proceso es parecido a crear una zona directa.





En el id de red pondremos la ip de nuestra red:



Por último le damos a finalizar



Lo único que falta es abrir la zona inversa y crear punteros como hemos hecho en la zona directa.

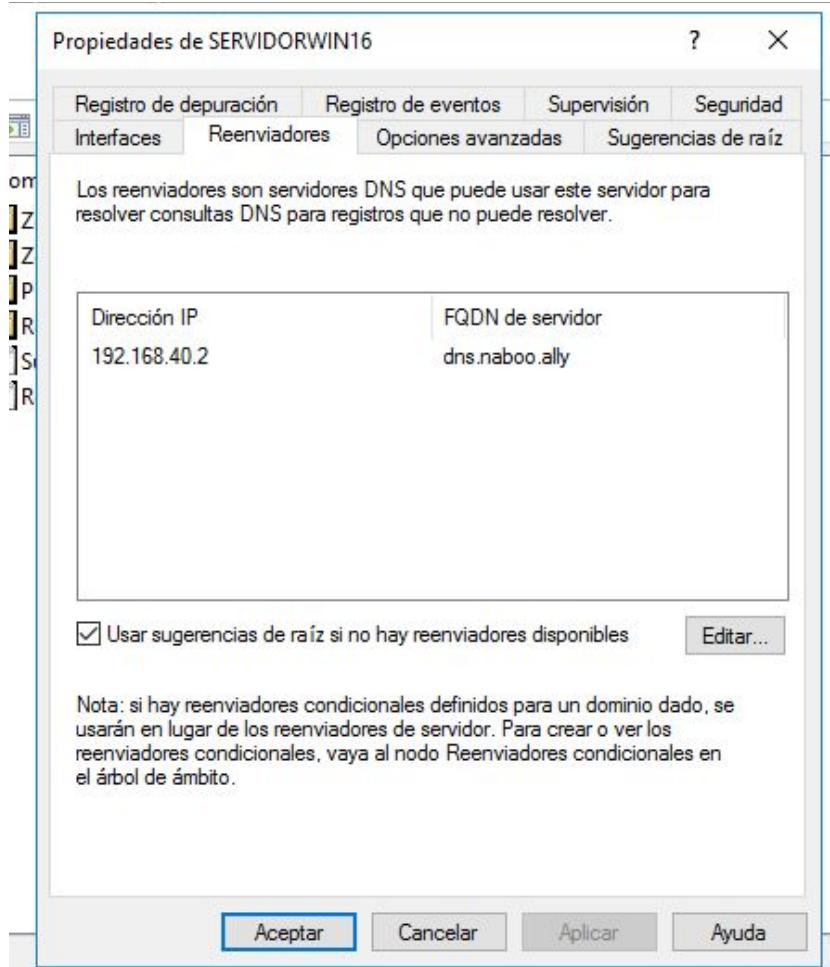
Administrador de DNS

Archivo Acción Ver Ayuda

Nombre	Tipo	Datos	Marca de tiempo
(igual que la carpeta principal)	Inicio de autoridad (SOA)	[7], servidorwin16.naboo.l...	static
(igual que la carpeta principal)	Servidor de nombres (NS)	servidorwin16.naboo.local.	static
192.168.4.136	Puntero (PTR)	winclient.naboo.local.	static
192.168.4.2	Puntero (PTR)	debianclient.naboo.local.	static
192.168.4.91	Puntero (PTR)	dhcp.naboo.local.	static
192.168.4.92	Puntero (PTR)	ServidorWin16.naboo.local.	static
192.168.4.93	Puntero (PTR)	proxy.naboo.local.	static

Una vez hecho esto el dns nos debería permitir hacer nslookup a un nombre de host o ip de nuestra red.

Una de las partes más importantes es establecer un reenviador de dns, esto lo podemos hacer dando clic derecho en el servidor y después en propiedades. Una vez aquí le damos a la pestaña de “Reenviadores” y aquí es donde debemos poner la ip de nuestro server dns de linux, en nuestro caso la 192.168.40.2.



Para poder usar las máquinas virtuales de windows desde Aulamix hemos usado el programa llamado “TeamViewer” que te permite conectarte a cualquier pc que tenga instalado el programa instalado sin tener que estar en lan.

Comprobando si el dns funciona correctamente:

```

> ad.naboo.local
Server:          192.168.4.92
Address:         192.168.4.92#53

Name:  ad.naboo.local
Address: 192.168.4.92
> 192.168.4.92
92.4.168.192.in-addr.arpa      name = ServidorWin16.naboo.local.
> dns.alderaan.ally
Server:          192.168.4.92
Address:         192.168.4.92#53

Non-authoritative answer:
Name:  dns.alderaan.ally
Address: 192.168.30.3
> 192.168.30.3
3.30.168.192.in-addr.arpa      name = dns.alderaan.ally.

Authoritative answers can be found from:
>

```

Webgraphy:

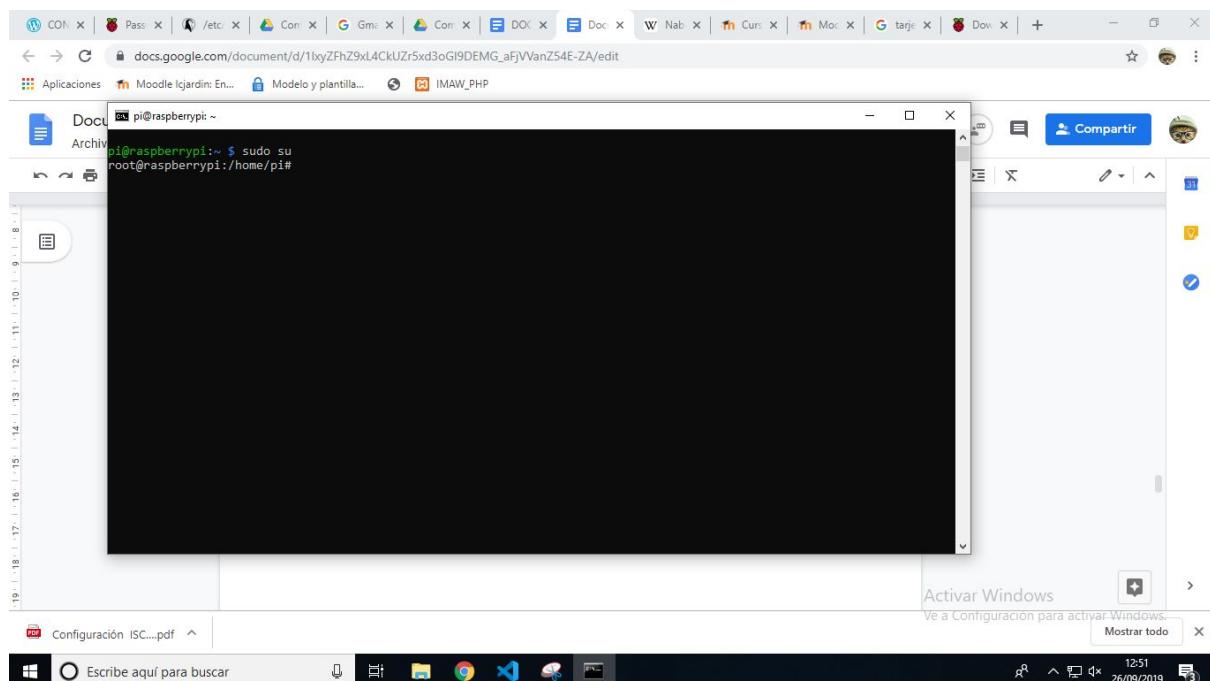
<https://www.solvetic.com/tutoriales/article/3284-instalar-y-configurar-servidor-dns-windows-server-2016/>

Raspberry PI:

The aim of this small-board computer is to use the DHCP protocol to give a dynamic IP to another computers.

Installation of the OS

For the configuration of the Raspberry Pi, we have to give format to the SD card because we are going to configure it with its default settings. When we have finished, we have to flash a Raspbian.iso (You can download it from their webpage) to the SD card so that we can configure it graphically. Then, with a HDMI cable and a keyboard you can begin with the installation graphically. The main username is “pi” and the password is “raspberry”. It is going to be connected to the switch B, and this switch is connected to the router to the 192.168.4.1 network adapter, acquiring network connection to the internet from the main network 172.20.202.1. Then we log on with the root user with the command sudo su and we are going to be able to set up the network configuration, opening the /etc/network/interfaces file.



Network configuration

In our case, the Raspberry Pi is going to be part of the naboo.local network, and its IP has to be part of the 192.168.4.0 network.

```
pi@raspberrypi: ~
GNU nano 3.2
/etc/network/interfaces

# interfaces(5) file used by ifup(8) and ifdown(8)

# Please note that this file is written to be used with dhcpcd
# For static IP, consult /etc/dhcpcd.conf and 'man dhcpcd.conf'

# Include files from /etc/network/interfaces.d
source-directory /etc/network/interfaces.d

auto lo
iface lo inet loopback

iface eth0 inet static
    address 192.168.4.91
    netmask 255.255.255.0
    gateway 192.168.4.1
    dns-nameservers 8.8.8.8
```

```
pi@dhcp: ~
root@dhcp:/home/pi# service networking restart
root@dhcp:/home/pi#
```

Naboo
Julen Fernández
Ibon Pina
SI2

This is just a comprobation that the raspberry connects properly to the internet via our router. We used the “ping” command.

```
pi@raspberrypi: ~
root@raspberrypi:/home/pi# ping 192.168.4.1
PING 192.168.4.1 (192.168.4.1) 56(84) bytes of data.
64 bytes from 192.168.4.1: icmp_seq=1 ttl=64 time=0.357 ms
64 bytes from 192.168.4.1: icmp_seq=2 ttl=64 time=0.239 ms
^C
--- 192.168.4.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 17ms
rtt min/avg/max/mdev = 0.239/0.298/0.357/0.059 ms
root@raspberrypi:/home/pi# ping 172.20.202.21
PING 172.20.202.21 (172.20.202.21) 56(84) bytes of data.
64 bytes from 172.20.202.21: icmp_seq=1 ttl=64 time=0.422 ms
64 bytes from 172.20.202.21: icmp_seq=2 ttl=64 time=0.260 ms
^C
--- 172.20.202.21 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 81ms
rtt min/avg/max/mdev = 0.260/0.341/0.422/0.081 ms
root@raspberrypi:/home/pi# ping 172.20.202.1
PING 172.20.202.1 (172.20.202.1) 56(84) bytes of data.
64 bytes from 172.20.202.1: icmp_seq=1 ttl=63 time=0.469 ms
64 bytes from 172.20.202.1: icmp_seq=2 ttl=63 time=0.386 ms
^C
--- 172.20.202.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 49ms
rtt min/avg/max/mdev = 0.386/0.427/0.469/0.046 ms
root@raspberrypi:/home/pi# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=51 time=9.77 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=51 time=9.83 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 2ms
rtt min/avg/max/mdev = 9.769/9.801/9.834/0.104 ms
root@raspberrypi:/home/pi#
```

DHCP configuration

After checking that the single-board computer can ping with the router and has access to the internet, we are going to install the service isc-dhcp-server. We can install it adding this command one prompt: apt-get install isc-dhcp-server. Remember that before we can install this service we have to change the /etc/apt/sources.list file. After installing it, we have to change this file /ETC/DEFAULT/ISC-DHCP-SERVER and add the network adapter which is connected to the net, in our case “eth0” as we can see it configured in the network interfaces file.

/etc/dhcp/dhcpd.conf

In this file we are going to add the range of IPs that we are going to have in our DHCP. We have to add the IP and the name of the domain. In our example the domain is naboo.local and the IP is the one of the DHCP server, the raspberry. The subnet is the network (192.168.4.0) and then we can add the ranges we need for our networks security. Here we have created 3 ranges to distribute the 200 devices. We have also reserved an IP for our Windows10 Client, which is 192.168.4.136. For this we should check the MAC address of that machine.

```

# dhcpcd.conf
#
# Sample configuration file for ISC dhcpcd
#
# option definitions common to all supported networks...
option domain-name "naboo.local";
option domain-name-servers 192.168.4.92;

default-lease-time 86000; #Tiempo por defecto de la concesión en segundos.
max-lease-time 172000; #Tiempo máximo de la concesión en segundos.

authoritative; #Establece el servidor como principal.
log-facility local7;

#Opciones de subred
subnet 192.168.4.0 netmask 255.255.255.0 { #Declaración de la subred.
range 192.168.4.2 192.168.4.90; #Primer rango de direcciones.
range 192.168.4.101 192.168.4.130; #Segundo rango de direcciones.
range 192.168.4.141 192.168.4.254; #Tercer rango de direcciones.
option routers 192.168.4.1; #Gateway (nuestro router).
option subnet-mask 255.255.255.0; #Máscara de subred.
option broadcast-address 192.168.4.255; #Dirección de broadcast
}
#Reserva de direcciones Ordenador Windows
host Windows 10 Client { #Declaramos la reserva
hardware ethernet 08:00:27:C2:78:2D; #Dirección MAC del dispositivo.
fixed-address 192.168.4.136; #Dirección IP reservada.
default-lease-time 172000;
}

dhcpcd -t /etc/dhcp/dhcpcd.conf

```

In order to check that the configuration is correct, we will use two virtual machines that are located in the PC SI1-03, which is connected to the B switch. One, with Debian, will have the first IP of the first range of the DHCP. The Windows machine will receive the reserved IP address

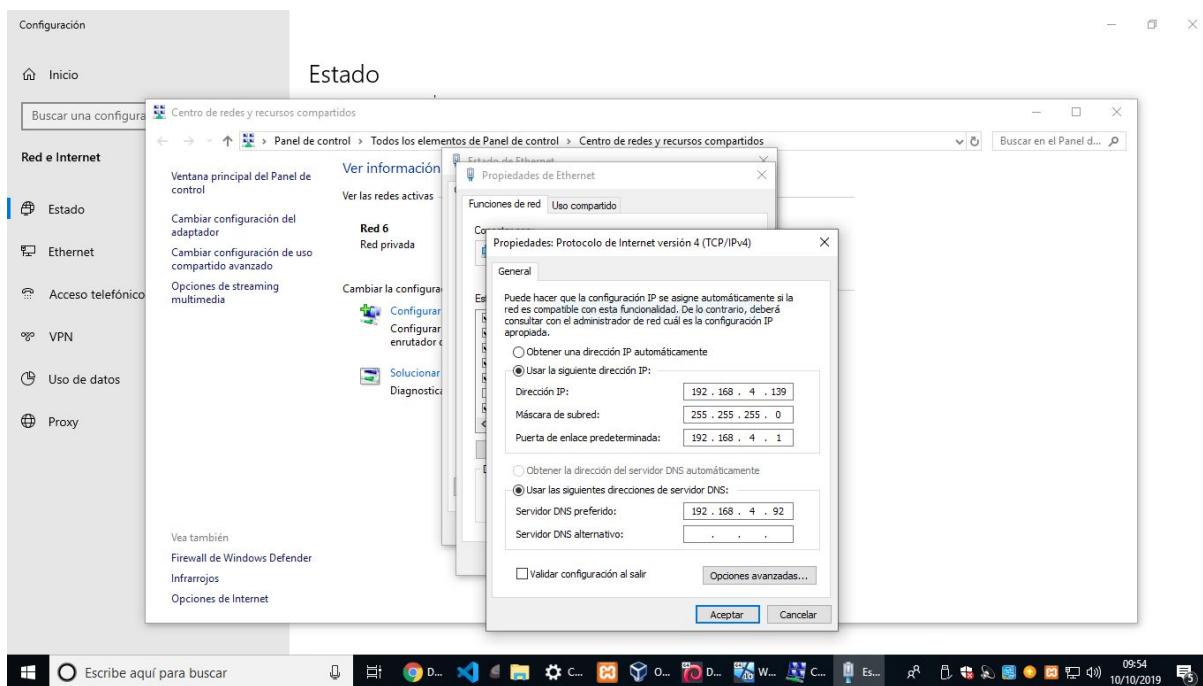
```

root@dhcp:/home/pi# service isc-dhcp-server status
● isc-dhcp-server.service - LSB: DHCP server
  Loaded: loaded (/etc/init.d/isc-dhcp-server; generated)
  Active: active (running) since Fri 2019-08-09 11:02:36 BST; 7s ago
    Docs: man:systemd-sysv-generator(8)
 Process: 571 ExecStart=/etc/init.d/isc-dhcp-server start (code=exited, status=0/SUCCESS)
   Tasks: 1 (limit: 2200)
  Memory: 6.0M
 CGroup: /system.slice/isc-dhcp-server.service
         └─590 /usr/sbin/dhcpcd -4 -q -cf /etc/dhcp/dhcpd.conf eth0

Aug 09 11:02:33 dhcp.naboo.local dhcpcd[590]: Internet Systems Consortium DHCP Server 4.4.1
Aug 09 11:02:33 dhcp.naboo.local dhcpcd[590]: Copyright 2004-2018 Internet Systems Consortium.
Aug 09 11:02:33 dhcp.naboo.local dhcpcd[590]: All rights reserved.
Aug 09 11:02:33 dhcp.naboo.local dhcpcd[590]: For info, please visit https://www.isc.org/software/dhcp/
Aug 09 11:02:33 dhcp.naboo.local dhcpcd[590]: Wrote 0 deleted host decls to leases file.
Aug 09 11:02:33 dhcp.naboo.local dhcpcd[590]: Wrote 0 new dynamic host decls to leases file.
Aug 09 11:02:33 dhcp.naboo.local dhcpcd[590]: Wrote 5 leases to leases file.
Aug 09 11:02:33 dhcp.naboo.local dhcpcd[590]: Server starting service.
Aug 09 11:02:36 dhcp.naboo.local isc-dhcp-server[571]: Starting ISC DHCPv4 server: dhcpcd.
Aug 09 11:02:36 dhcp.naboo.local systemd[1]: Started LSB: DHCP server.
root@dhcp:/home/pi#

```

The two machine clients used to test everything works properly are installed in the SI2-03 computer, which is going to have an IP inside the 192.168.4.0 network and is connected to Switch B.

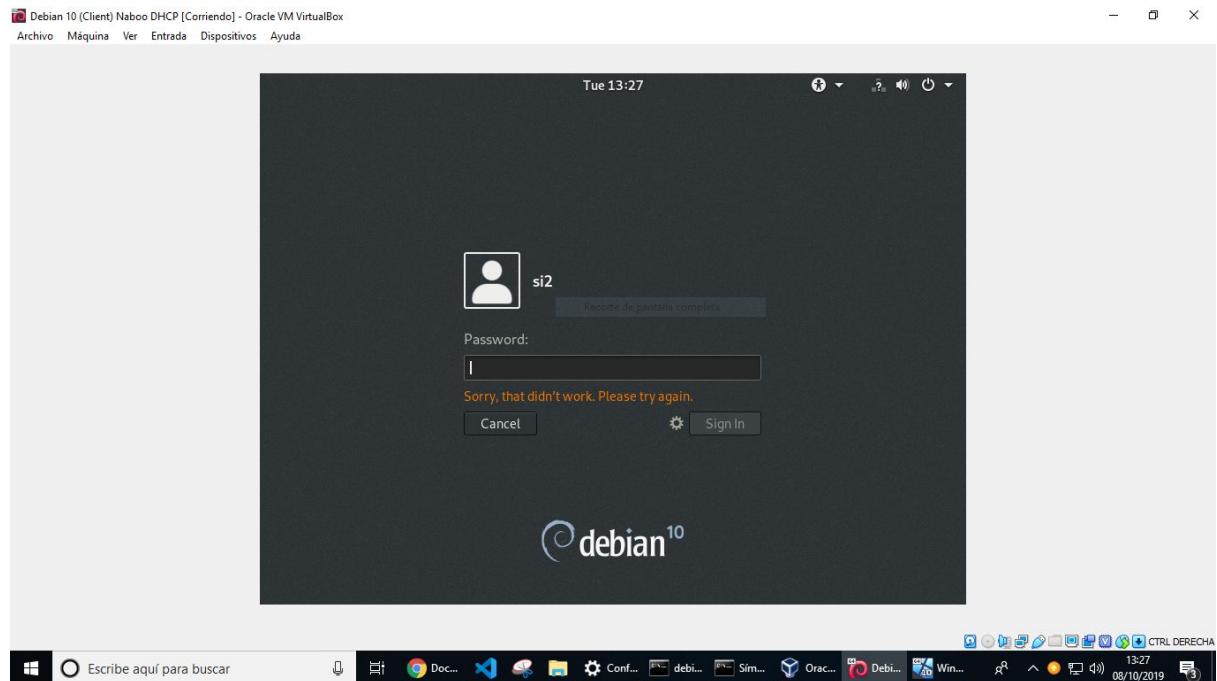


Both clients are going to be in bridge mode so that they have access to the internet.

Naboo
Julen Fernández
Ibon Pina
SI2

Debian 10 Client

This machine is going to have the graphic environment installed



Network configuration

Naboo
Julen Fernández
Ibon Pina
SI2

```

Debian 10 (Client) Naboo DHCP [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda

Activities Terminal Tue 13:30
si2@debiandcliente: ~
File Edit View Search Terminal Help
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
iface lo inet loopback

iface enp0s3 inet dhcp

[ Read 13 lines ]
[G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^M Replace ^U Uncut Text ^T To Spell ^L Go To Line

```

Escribe aquí para buscar

```

Debian 10 (Client) Naboo DHCP [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda

Activities Terminal Tue 13:31
root@debiandcliente:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 brd 0.0.0.0 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 brd 0.0.0.0 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
    link/ether 08:00:27:f4:00:95 brd ff:ff:ff:ff:ff:ff
    inet 192.168.4.2/24 brd 192.168.4.255 scope global dynamic enp0s3
        valid_lft 258848sec preferred_lft 258848sec
    inet6 fe80::a00:27ff:fe00:95/64 scope link
        valid_lft forever preferred_lft forever
root@debiandcliente:~#

```

Escribe aquí para buscar

Debian 10 Client to the domain

We will need to install a package from the internet. We can install it with this command:

`wget`

<https://github.com/BeyondTrust/pbis-open/releases/download/8.5.3/pbis-open-8.5.3.293.linux.x86.deb.sh>

https://github.com/BeyondTrust/pbis-open/releases/download/9.1.0/pbis-open-9.1.0.551.linux.x86_64.deb.sh

We will need to give the necessary permissions in order to be able to execute it

```
chmod +x pbis-open-8.5.3.293.linux.x86_64.deb.sh
```

Then we execute it.

```
sudo ./pbis-open-8.5.3.293.linux.x86_64.deb.sh
```

Finally, we have to write this command:

domainjoin.cli join (domain) (user with privileges from the domain).

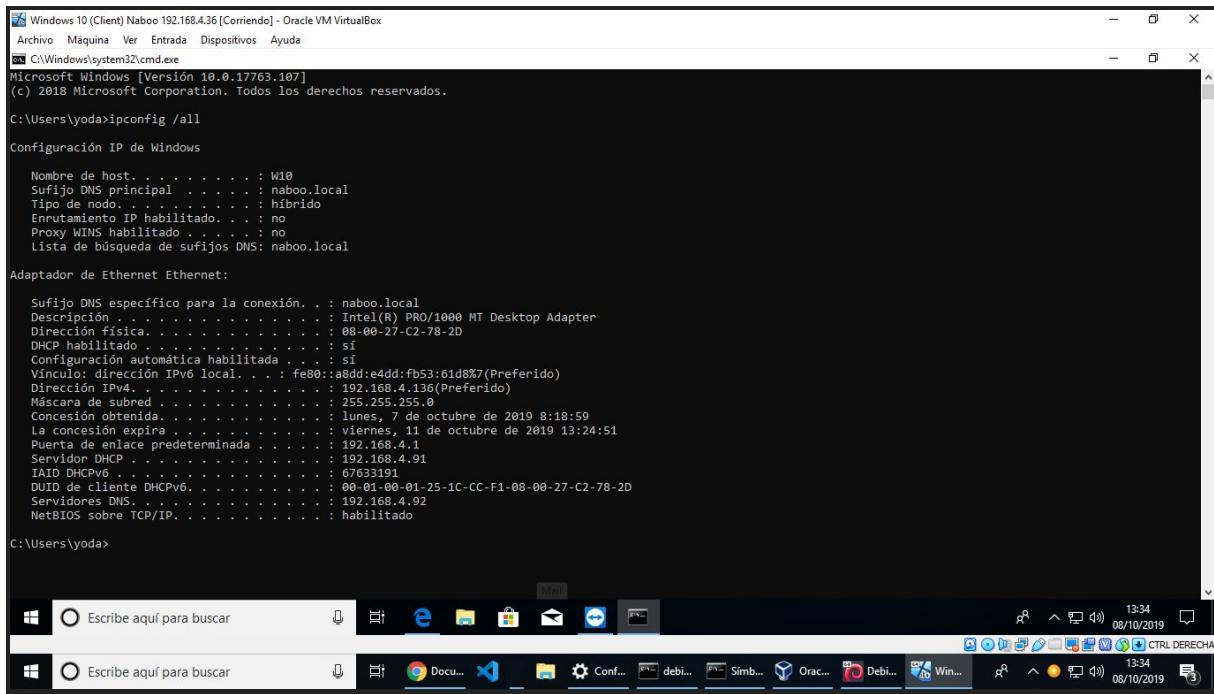
```
C:\Users\si2>ssh si2@192.168.4.2
Password:
Linux debiancliente.naboo.local 4.19.0-6-amd64 #1 SMP Debian 4.19.67-2 (2019-08-28-28) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Oct  8 13:46:19 2019 from 192.168.40.133
si2@debiancliente:~$ su -
Password:
root@debiancliente:~# domainjoin-cli join naboo.local administrador@naboo.local
```

Windows 10 Client

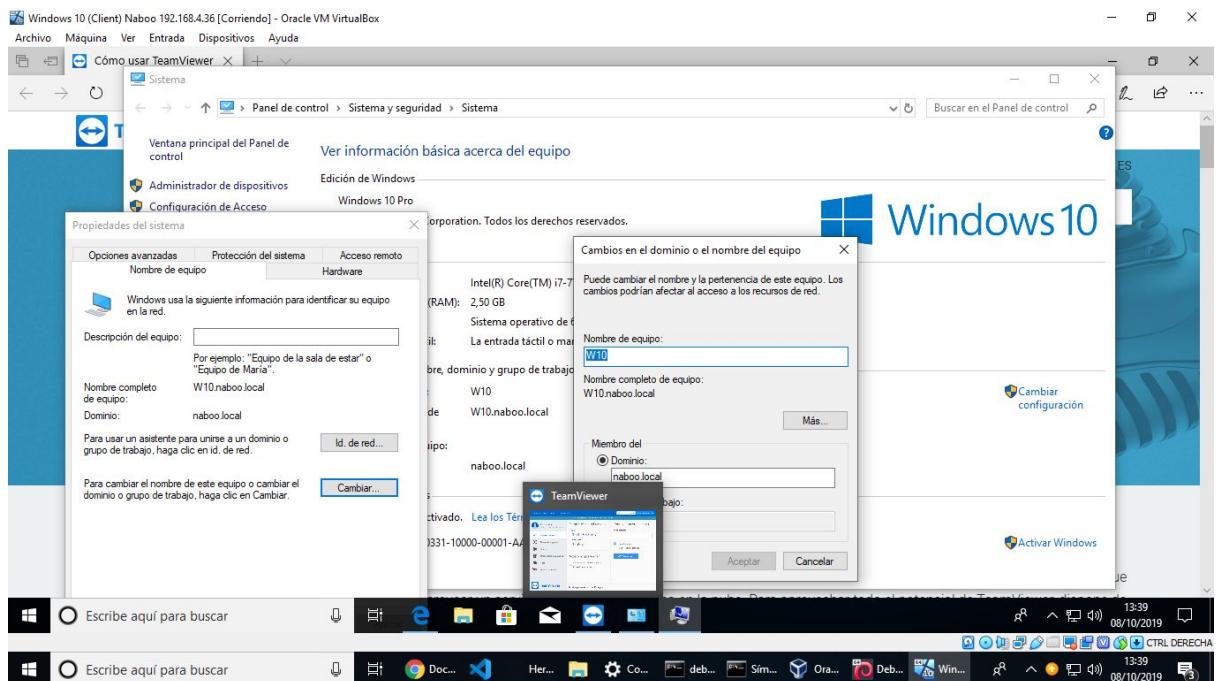
Network configuration



Windows 10 (Client) Naboo 192.168.4.36 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
cmd C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 10.0.17763.107]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.
C:\Users\yoda>ipconfig /all
Configuración IP de Windows
Nombre de host : W10
Sufijo DNS principal : naboo.local
Tipo de nodo : híbrido
Enrutamiento IP habilitado : no
Proxy WINS habilitado : no
Lista de búsqueda de sufijos DNS: naboo.local
Adaptador de Ethernet Ethernet:
Sufijo DNS específico para la conexión. : naboo.local
Descripción : Intel(R) PRO/1000 MT Desktop Adapter
Dirección física : 08-00-27-C2-78-00
DHCP habilitado : sí
Configuración automática habilitada : sí
Vínculo: dirección IPv6 local. : fe80::a8dd:e4dd:fb53:61d8%7(Preferido)
Dirección IPv4 : 192.168.4.136(Preferido)
Máscara de subred : 255.255.255.0
Concesión obtenida : lunes, 7 de octubre de 2019 8:18:59
La concesión expira : viernes, 11 de octubre de 2019 13:24:51
Puerta de enlace predeterminada : 192.168.4.1
Servidor DHCP : 192.168.4.91
IAID DHCPv6 : 67633191
DUID de cliente DHCPv6 : 00-01-00-01-25-1C-CC-F1-08-00-27-C2-78-2D
Servidores DNS : 192.168.4.92
NetBIOS sobre TCP/IP. : habilitado
C:\Users\yoda>

Add Windows 10 Client to the domain

Once we have the IP configured by the DHCP, we will add the machine to the domain and we have to enter the panel control and system. Here we will enter advanced configuration and here we can change the domain, naboo.local in our case.



```

Windows 10 (Client) Naboo 192.168.4.36 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 10.0.17763.107]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Users\yoda>ipconfig /all

Configuración IP de Windows

Nombre de host . . . . . : W10
Sufijo DNS principal . . . . . : naboo.local
Tipo de nodo . . . . . : híbrido
Enrutamiento IP habilitado . . . . . : no
Proxy WINS habilitado . . . . . : no
Lista de búsqueda de sufijos DNS: naboo.local

Adaptador de Ethernet Ethernet:

Sufijo DNS específico para la conexión . . . . . : naboo.local
Descripción . . . . . : Intel(R) PRO/1000 MT Desktop Adapter
Dirección física . . . . . : 08-00-27-C2-78-2D
DHCP habilitado . . . . . : sí
Configuración automática habilitada . . . . . : sí
Vínculo: dirección IPv6 local. . . . . : fe80::a8dd:e4dd:fb53:61d8%7(Preferido)
Dirección IPv4 . . . . . : 192.168.4.136(Preferido)
Máscara de subred . . . . . : 255.255.255.0
Concesión obtenida . . . . . : lunes, 7 de octubre de 2019 8:18:59
La concesión expira . . . . . : viernes, 11 de octubre de 2019 13:24:51
Puerta de enlace predeterminada . . . . . : 192.168.4.1
Servidor DHCP . . . . . : 192.168.4.91
IAID DHCPv6 . . . . . : 67633191
DUID de cliente DHCPv6 . . . . . : 00-01-00-01-25-1C-CC-F1-08-00-27-C2-78-2D
Servidores DNS . . . . . : 192.168.4.92
NetBIOS sobre TCP/IP. . . . . : habilitado

C:\Users\yoda>

```

Webgraphy:

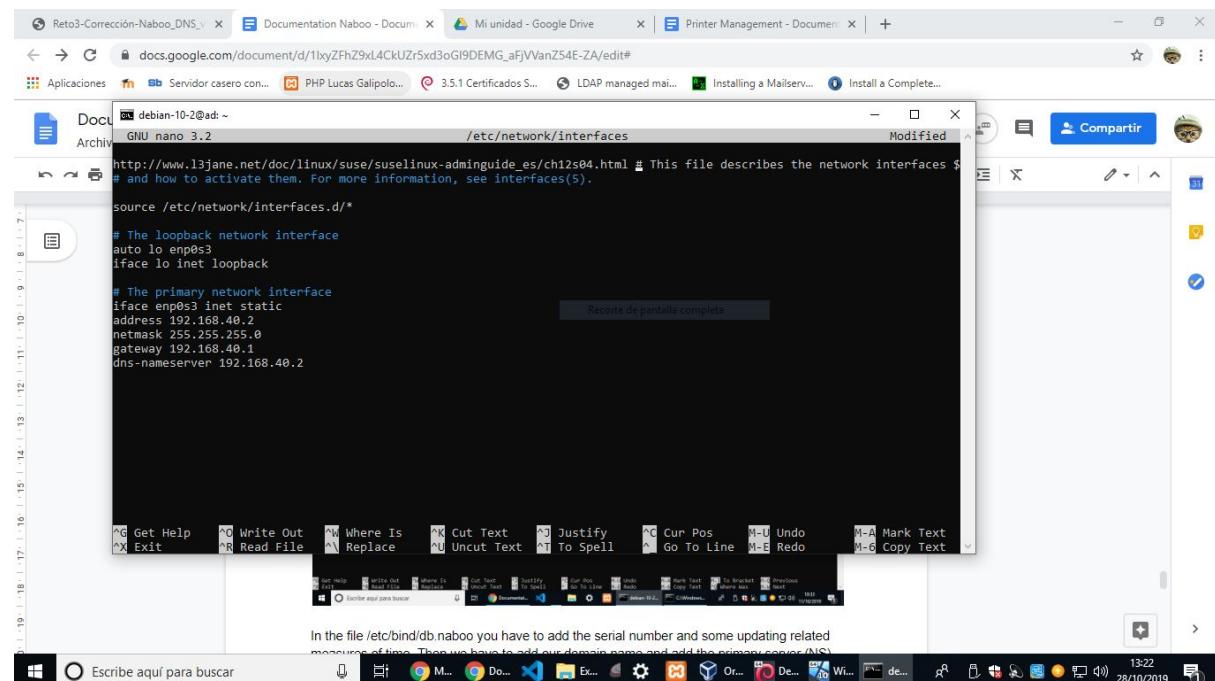
<https://sobrebits.com/montar-un-servidor-casero-con-raspberry-pi-parte>
<https://www.itprotoday.com/windows-78/how-can-i-join-domain-comma>
<nd-line-3-configurar-servidor-dhcp/>

Naboo.ally configuration

For the configuration of the machines in naboo.ally we use the SI1-04 PC with 192.168.40.140 IP. It is going to be connected to switch A which is in 192.168.40.0 network.

Linux DNS Configuration

For the DNS configuration, we will use a Debian machine with 192.168.40.2 IP and the Bind9 service.



In order to install the service we should use `apt-get install bind9 bind9-doc dnsutils`.

First of all, we are going to add the zones that we are going to use, one for the domain and the other for the reverse resolution in `/etc/bind/named.conf` file.

Our domain is `naboo.ally` and we have to establish the configuration file .

In the reverse resolution you have to add the network, and you have to declare it upside down without the last 8 bits and with `in-addr-arpa` and then add the configuration file.

We should check that the syntax of the configuration file is correct, with this command can be done `named-checkconf /etc/named.conf`.

```

GNU nano 3.2                               /etc/bind/named.conf.local
Archivo named.conf: Archivo principal de configuración.
Archivo named.conf.options: Opciones genéricas.
Archivo named.conf.local: Especificación particular de este servidor DNS.
Archivo db.127: Especificación dirección de retorno.// 
// Do any local configuration here
// 

// Consider adding the 1918 zones here, if they are not used in your
// organization
#include "/etc/bind/zones.rfc1918";
#Zona del dominio
zone "naboo.ally" {
    type master;
    file "/etc/bind/db.naboo";
},
#Zona de resolución inversa
zone "48.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/db.192.168.40";
};


```

In the file /etc/bind/db.naboo you have to add the serial number and some updating related measures of time. Then we have to add our domain name and add the primary server (NS). To add a component of the zone we have to add a name, and its address (A). If a single host has different names, we can use CNAME to add more than one DNS names referring to the same machine.

```

$TTL 604800
@ IN SOA dns.naboo.ally. root.naboo.ally. (
2019071001 ; Serial
604800 ; Refresh
86400 ; Retry
2419200 ; Expire
604800 ) ; Default TTL

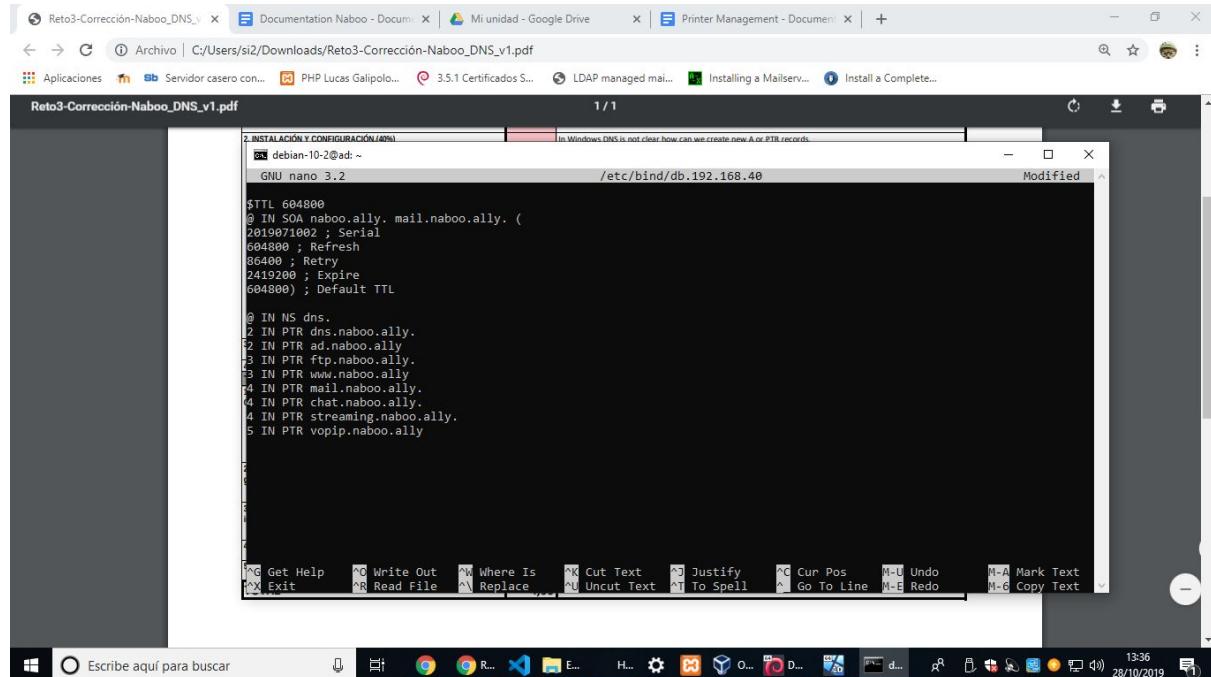
IN NS dns.naboo.ally.
mail IN MX 10 mail.naboo.ally.
dns IN A 192.168.40.2
ad IN CNAME dns
mail IN A 192.168.40.4
chat IN CNAME mail
streaming IN CNAME mail
ftp IN A 192.168.40.3
www IN CNAME ftp
voip IN A 192.168.40.5


```

The file for the reverse resolution is going to be named /etc/bind/db.192.168.40 (Our network). As in the domain zone you have to add the serial number and some updating related measures of time. Then you have to add the IPs that will form the zone. First one

Naboo
Julen Fernández
Ibon Pina
SI2

with @ is the name of the DNS server. The other ones must start by the last 8 bits of its address, 2, 3, 4 and 5 in our case (192.168.40.2, 192.168.40.3, 192.168.40.4, 192.168.40.5) and is recommendable to add the name of the machine.



```
$TTL 604800
@ IN SOA naboo.ally. mail.naboo.ally. (
2019071802 ; Serial
604800 ; Refresh
86400 ; Retry
2419200 ; Expire
604800 ) ; Default TTL

@ IN NS dns.
2 IN PTR dns.naboo.ally.
2 IN PTR ad.naboo.ally.
3 IN PTR ftp.naboo.ally.
3 IN PTR www.naboo.ally.
4 IN PTR mail.naboo.ally.
4 IN PTR chat.naboo.ally.
4 IN PTR streaming.naboo.ally.
5 IN PTR vopip.naboo.ally
```

To check the configuration of the database of the zones is correct we can use this command:

```
named-checkzone naboo.local /etc/bind/db.naboo
```

(Where naboo.local is the domain and /etc/bind/db.naboo the configuration file)

```
named-checkzone naboo.local /etc/bind/db.192.168.40
```

As the coruscant.capital dns is configured, we are going to use it as a forwarder. The forwarder forwards DNS queries for external DNS names to DNS servers outside our network.

```

debian-10-2@dns: ~
GNU nano 3.2                               /etc/bind/named.conf.options

options {
    directory "/var/cache/bind";

    // If there is a firewall between you and nameservers you want
    // to talk to, you may need to fix the firewall to allow multiple
    // ports to talk. See http://www.kb.cert.org/vuls/id/800113

    // If your ISP provided one or more IP addresses for stable
    // nameservers, you probably want to use them as forwarders.
    // Uncomment the following block, and insert the addresses replacing
    // the all-0's placeholder.

    forwarders {
        172.20.202.251;
        8.8.8.8;
    };

    =====
    // If BIND logs error messages about the root key being expired,
    // you will need to update your keys. See https://www.isc.org/bind-keys
    =====
    dnssec-validation no;
    listen-on-v6 { any; };
    recursion yes;
    allow-query{ any; };
};

Activar Windows
Ve a Configuración para activar Windows.
M-F1 To Bracket M-Q Previous
M-Q Where Was M-N Next
11:58
18/10/2019

```

We have also configured the local DNS. You must change `/etc/network/interfaces` and `/etc/resolv.conf` in order to work everything properly adding the nameserver of the DNS (The IP itself) and the domain name.

```

debian-10-2@dns: ~
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
iface lo inet loopback

# The primary network interface
iface enp0s3 inet static
    address 192.168.40.2
    netmask 255.255.255.0
    gateway 192.168.40.1
    dns-nameserver 192.168.40.2

Activar Windows
Ve a Configuración para activar Windows.
M-F1 To Bracket M-Q Previous
M-Q Where Was M-N Next
13:07
11/10/2019

```

debian-10-2@dns: ~

GNU nano 3.2

/etc/resolv.conf

```
nameserver 192.168.40.2
search naboo.ally
```

Recorte de pantalla completa

Get Help Write Out Where Is Cut Text Justify Cur Pos To Spell Undo Mark Text To Bracket Previous

Exit Read File Replace Uncut Text To To Line Undo Redo Copy Text Where Was Next

Escribe aquí para buscar

Once we have finished the whole configuration, we are going to restart the service and check its status.

```
root@ad:~# service bind9 restart
root@ad:~# service bind9 status
● bind9.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/bind9.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2019-10-28 08:27:09 EDT; 1s ago
     Docs: man:bind9(8)
   Process: 656 ExecStart=/usr/sbin/named $OPTIONS (code=exited, status=0/SUCCESS)
 Main PID: 657 (named)
   Tasks: 4 (limit: 1150)
    Memory: 15.6M
      CGroup: /system.slice/bind9.service
              └─ 657 /usr/sbin/named -u bind

Oct 28 08:27:09 ad named[657]: managed-keys-zone/externa: loaded serial 0
Oct 28 08:27:09 ad named[657]: zone naboo.ally/IN/interna: loaded serial 2019071001
Oct 28 08:27:09 ad named[657]: zone 40.168.192.in-addr.arpa/IN/interna: loaded serial 2019071002
Oct 28 08:27:09 ad named[657]: zone 202.20.172.in-addr.arpa/IN/externa: loaded serial 2019071002
Oct 28 08:27:09 ad named[657]: zone naboo.ally/IN/externa: loaded serial 2019071001
Oct 28 08:27:09 ad named[657]: all zones loaded
Oct 28 08:27:09 ad systemd[1]: Started BIND Domain Name Server.
Oct 28 08:27:09 ad named[657]: running
Oct 28 08:27:09 ad named[657]: zone 40.168.192.in-addr.arpa/IN/interna: sending notifies (serial 2019071002)
Oct 28 08:27:09 ad named[657]: zone 202.20.172.in-addr.arpa/IN/externa: sending notifies (serial 2019071002)
root@ad:~#
```

Some checkings to proof this configuration works with nslookup.

```

si2@proxy:~$ nslookup
> www.naboo.ally
Server:      192.168.4.92
Address:    192.168.4.92#53

Non-authoritative answer:
Name:  www.naboo.ally
Address: 192.168.40.2
> dns.naboo.ally
Server:      192.168.4.92
Address:    192.168.4.92#53

Non-authoritative answer:
Name:  dns.naboo.ally
Address: 192.168.40.2
> 192.168.4.92
92.4.168.192.in-addr.arpa      name = ServidorWin16.naboo.local.
> ^Csi2@proxy:~$ nslookup
> dhcp.naboo.local
Server:      192.168.4.92
Address:    192.168.4.92#53

Name:  dhcp.naboo.local
Address: 192.168.4.91
> 192.168.4.91
91.4.168.192.in-addr.arpa      name = dhcp.naboo.local.
>

debian-10-2@ad:~$ nslookup 192.168.40.2
2.40.168.192.in-addr.arpa      name = ad.naboo.ally.
2.40.168.192.in-addr.arpa      name = dns.naboo.ally.

debian-10-2@ad:~$ nslookup 192.168.40.3
3.40.168.192.in-addr.arpa      name = www.naboo.ally.
3.40.168.192.in-addr.arpa      name = ftp.naboo.ally.

debian-10-2@ad:~$ nslookup 192.168.40.4
4.40.168.192.in-addr.arpa      name = mail.naboo.ally.
4.40.168.192.in-addr.arpa      name = chat.naboo.ally.
4.40.168.192.in-addr.arpa      name = streaming.naboo.ally.

debian-10-2@ad:~$ nslookup ad.naboo.ally
Server:      192.168.40.2
Address:    192.168.40.2#53

ad.naboo.ally canonical name = dns.naboo.ally.
Name:  dns.naboo.ally
Address: 192.168.40.2

debian-10-2@ad:~$ nslookup www.naboo.ally
Server:      192.168.40.3
Address:    192.168.40.3#53

www.naboo.ally canonical name = ftp.naboo.ally.
Name:  ftp.naboo.ally
Address: 192.168.40.3

debian-10-2@ad:~$ nslookup chat.naboo.ally
Server:      192.168.40.4
Address:    192.168.40.4#53

chat.naboo.ally canonical name = mail.naboo.ally.
Name:  mail.naboo.ally
Address: 192.168.40.4

debian-10-2@ad:~$
```

Webgraphy

<https://www.linuxtechi.com/install-configure-bind-9-dns-server-ubuntu-debian/>

<https://lcaballero.wordpress.com/2017/07/11/instalar-dns-bind9-en-ubuntu-15-10/>

Issues during the configuration

The first problem we had to resolve was the inability to add services to linux machines because the repositories were not working properly and we had to copy ones from the internet.

Another problem we have had during the router configuration was the running-up script. It was not working at first because we forgot to add this command: update-rc.d script name defaults.

The most important problem that we have had is that we can't make the ssh work with a passphrase, we have some screenshots the first time that we did it where it works. But it currently doesn't work.

Tareas Reto 4

1 Close the firewall **Sin acabar**

the first thing we must do is to “close” the firewall of each base using iptables. we must avoid all the traffic through the firewall, except for those communications that are absolutely necessary for a good system performance.

2 Keep a record of the access tries

In order to control these attacks, we will keep a record of the access tries (both successful and unsuccessful) to the firewall.

3 Install and configure a LDAP server in naboo.ally **Crear usuarios**

we will install and configure a LDAP server in xxxx.ally (OpenLDAP), in which we will create the users (with their respective passwords) and groups following the same structure we have used in the Active Directory of Windows 2016 Server on xxxx.local network.

In order to manage our network properly we are going to set up open-ldap so that we can have users and groups. Once we have each machine with the correct network configuration we are going to use the mail virtual machine that uses the ip 192.168.40.4 for the installation and configuration of ldap.

The first step is to install the ldap packets. We can do that using the command
apt-get install slapd ldap-utils ldapscripts

```
debian-10-2@dns:~$ su -
Password:
root@dns:~# apt-get update
Hit:1 http://ftp.es.debian.org/debian buster InRelease
Get:2 http://ftp.es.debian.org/debian buster-updates InRelease [49.3 kB]
Get:3 http://security.debian.org/debian-security buster/updates InRelease [39.1 kB]
Get:4 http://security.debian.org/debian-security buster/updates/main Sources [77.2 kB]
Get:5 http://security.debian.org/debian-security buster/updates/main amd64 Packages [99.9 kB]
Get:6 http://security.debian.org/debian-security buster/updates/main Translation-en [62.0 kB]
Fetched 327 kB in 13s (25.1 kB/s)
Reading package lists... Done
root@dns:~# apt-get install slapd ldap-utils
```

During the installation of this packet the configuration of slpad will pop up.

Naboo
Julen Fernández
Ibon Pina
SI2

Here we will configure the ldap administrator's password, the domain name, and afterwards the installation will finish.

The terminal window has a blue header bar with the title 'Configuring slapd'. The main area is grey with black text. The first screenshot shows the password entry step. The second shows the domain name entry. The third shows the database backend selection.

```
Configuring slapd
Please enter the password for the admin entry in your LDAP directory.

Administrator password:
*****
<Ok>

Configuring slapd
The DNS domain name is used to construct the base DN of the LDAP directory.
For example, 'foo.example.org' will create the directory with 'dc=foo,
dc=example, dc=org' as base DN.

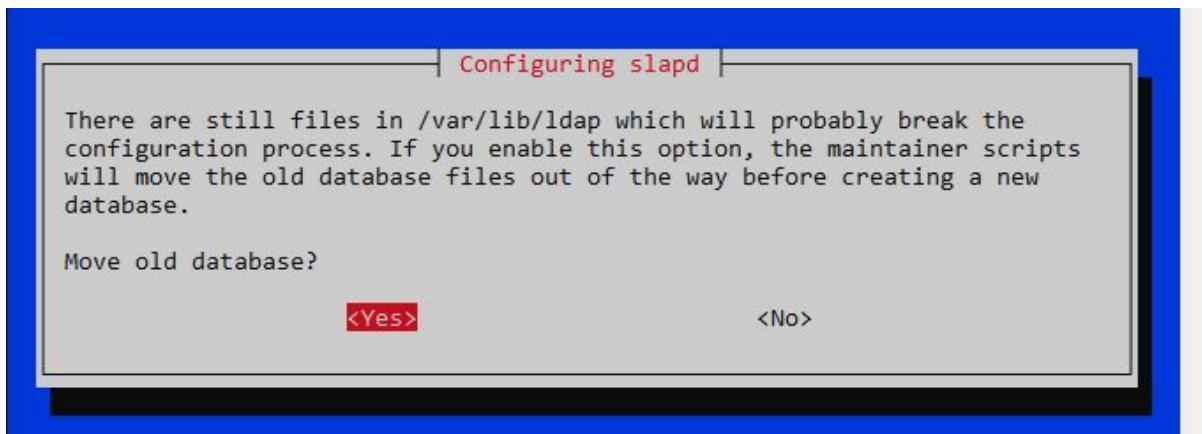
DNS domain name:
naboo.ally
<Ok>

Configuring slapd
HDB and BDB use similar storage formats, but HDB adds support for subtree
renames. Both support the same configuration options.

The MDB backend is recommended. MDB uses a new storage format and requires
less configuration than BDB or HDB.

In any case, you should review the resulting database configuration for
your needs. See /usr/share/doc/slapd/README.Debian.gz for more details.

Database backend to use:
BDB
HDB
MDB
<Ok>
```



Once we install the service we can start entering groups, organizational units and users to open ldap. For this we will just create a file called “*.ldif”, here we are going to write the proper entries for the group/ou/user we want to create.

Using the **slapcat** command we can see the actual configuration and entries in our ldap:

```
debian-10-2@dns: ~
dn: dc=naboo,dc=ally
objectClass: top
objectClass: dcObject
objectClass: organization
o: nabooallyorg
dc: naboo
structuralObjectClass: organization
entryUUID: 5bc8e80a-89a8-1039-8e69-77f91d63e5b4
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191023061613Z
entryCSN: 20191023061613.491136Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191023061613Z

dn: cn=admin,dc=naboo,dc=ally
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: LDAP administrator
userPassword:: e1NTSEF9MEFqN21kQ3piaGgwbERGNW0yd0hHbkIyVEV3K0RBSE4=
structuralObjectClass: organizationalRole
entryUUID: 5bc9590c-89a8-1039-8e6a-77f91d63e5b4
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191023061613Z
entryCSN: 20191023061613.494059Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191023061613Z
```

This is all the added groups and users:

These two organizational units are compulsory for the ldap to work properly.

```
dn: ou=groups,dc=naboo,dc=ally
objectClass: top
objectClass: organizationalUnit
ou: groups
description: generic groups branch
structuralObjectClass: organizationalUnit
entryUUID: 829d2648-89ae-1039-9005-db09c151901e
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191023070015Z
entryCSN: 20191023070015.615911Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191023070015Z

dn: ou=people,dc=naboo,dc=ally
objectClass: top
objectClass: organizationalUnit
ou: people
description: generic users branch
structuralObjectClass: organizationalUnit
entryUUID: 829dde94-89ae-1039-9006-db09c151901e
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191023070015Z
entryCSN: 20191023070015.620638Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191023070015Z
```

Created groups:

```
dn: cn=Usuarios,ou=people,dc=naboo,dc=ally
objectClass: top
objectClass: posixGroup
gidNumber: 2000
cn: Usuarios
memberUid: yoda
structuralObjectClass: posixGroup
entryUUID: 9945f5fa-89ae-1039-9007-db09c151901e
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191023070053Z
entryCSN: 20191023070053.632116Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191023070053Z

dn: cn=AdministratorUsers,ou=groups,dc=naboo,dc=ally
objectClass: top
objectClass: posixGroup
cn: AdministratorUsers
memberUid: kenobi
structuralObjectClass: posixGroup
entryUUID: 994692b2-89ae-1039-9008-db09c151901e
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191023070053Z
gidNumber: 10001
entryCSN: 20191107221530.095816Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191107221530Z

dn: cn=Generals,ou=groups,dc=naboo,dc=ally
objectClass: top
objectClass: posixGroup
cn: Generals
structuralObjectClass: posixGroup
entryUUID: 994713e0-89ae-1039-9009-db09c151901e
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191023070053Z
memberUid: carlissian
gidNumber: 10003
entryCSN: 20191107221227.711666Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191107221227Z
```

```
dn: cn=Captains,ou=groups,dc=naboo,dc=ally
objectClass: top
objectClass: posixGroup
cn: Captains
structuralObjectClass: posixGroup
entryUUID: 99479a04-89ae-1039-900a-db09c151901e
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191023070053Z
memberUid: dreis
gidNumber: 10002
entryCSN: 20191107221213.433004Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191107221213Z

dn: cn=Pilots,ou=groups,dc=naboo,dc=ally
objectClass: top
objectClass: posixGroup
cn: Pilots
structuralObjectClass: posixGroup
entryUUID: 9947f076-89ae-1039-900b-db09c151901e
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191023070053Z
memberUid: bridger
memberUid: creel
memberUid: farrell
memberUid: goban
gidNumber: 10004
entryCSN: 20191107221234.319620Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191107221234Z
```

Created users:

```
dn: uid=yoda,ou=people,dc=naboo,dc=ally
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
objectClass: person
uid: yoda
uidNumber: 2001
homeDirectory: /home/nfs/yoda
loginShell: /bin/bash
userPassword:: e1NTSEF9Nk10M2c0a2VqUDU0ei8yd00xQjBjUmdKM0lQVzNsN0E=
mail: yoda@gmail.com
givenName: yoda
structuralObjectClass: inetOrgPerson
entryUUID: f850f1f8-89ae-1039-900c-db09c151901e
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191023070333Z
sn: yoda
cn: yoda
gidNumber: 10001
entryCSN: 20191107222407.929791Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191107222407Z
```

```
dn: uid=kenobi,ou=people,dc=naboo,dc=ally
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
objectClass: person
cn:: a2Vub2JpCg==
uid: kenobi
uidNumber: 2002
homeDirectory: /home/nfs/kenobi
loginShell: /bin/bash
userPassword:: e1NTSEF9Nkl0M2c0a2VqUDU0ei8yd00xQjBjUmdKM0lQVzNsN0E=
mail: kenobi@gmail.com
givenName: kenobi
structuralObjectClass: inetOrgPerson
entryUUID: 2a7fb61c-8ace-1039-9873-db2ba338dcb9
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191024171922Z
sn: kenobi
gidNumber: 10001
entryCSN: 20191107222359.033816Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191107222359Z
```

```
dn: uid=carlissian,ou=people,dc=naboo,dc=ally
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
objectClass: person
cn:: Y2FybGlzc2lhbg==
uid: carlissian
uidNumber: 2003
homeDirectory: /home/nfs/carlissian
loginShell: /bin/bash
userPassword:: e1NTSEF9Nkl0M2c0a2VqUDU0ei8yd00xQjBjUmdKM0lQVzNsN0E=
sn:: Y2FybGlzc2lhbg==
mail: carlissian@gmail.com
givenName: carlissian
structuralObjectClass: inetOrgPerson
entryUUID: 2648bd2c-8acf-1039-895f-bfce5da7c238
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191024172625Z
gidNumber: 10003
entryCSN: 20191107222328.025891Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191107222328Z
```

```
dn: uid=dreis,ou=people,dc=naboo,dc=ally
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
objectClass: person
cn:: ZHJlaXMK
uid: dreis
uidNumber: 2004
homeDirectory: /home/nfs/dreis
loginShell: /bin/bash
userPassword:: e1NTSEF9Nkl0M2c0a2VqUDU0ei8yd00xQjBjUmdKM0lQVzNsN0E=
sn:: ZHJlaXMK
mail: dreis@gmail.com
givenName: dreis
structuralObjectClass: inetOrgPerson
entryUUID: a6b32c24-8dbb-1039-9e6d-990db7be4cc7
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191028104424Z
gidNumber: 10002
entryCSN: 20191107222345.097725Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191107222345Z
```

```
dn: uid=bridger,ou=people,dc=naboo,dc=ally
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
objectClass: person
uid: bridger
homeDirectory: /home/nfs/bridger
loginShell: /bin/bash
userPassword:: e1NTSEF9Nkl0M2c0a2VqUDU0ei8yd00xQjBjUmdKM0lQVzNsN0E=
mail: bridger@gmail.com
givenName: bridger
structuralObjectClass: inetOrgPerson
entryUUID: 5402fe9e-8dbd-1039-8afb-f955b50e6147
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191028105624Z
uidNumber: 2006
cn: bridger
sn: bridger
gidNumber: 10004
entryCSN: 20191107222437.161734Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191107222437Z
```

```
dn: uid=creel,ou=people,dc=naboo,dc=ally
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
objectClass: person
cn:: Y3JlZWwK
uid: creel
uidNumber: 2005
homeDirectory: /home/nfs/creel
loginShell: /bin/bash
userPassword:: e1NTSEF9Nkl0M2c0a2VqUDU0ei8yd00xQjBjUmdKM0lQVzNsN0E=
sn:: Y3JlZWwK
mail: creel@gmail.com
givenName: creel
structuralObjectClass: inetOrgPerson
entryUUID: 09cba0aa-8dbe-1039-8afc-f955b50e6147
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191028110129Z
gidNumber: 10004
entryCSN: 20191107222226.553815Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191107222226Z
```

```
dn: uid=farrell,ou=people,dc=naboo,dc=ally
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
objectClass: person
cn:: ZmFycmVsbaO=
uid: farrell
uidNumber: 2007
homeDirectory: /home/nfs/farrell
loginShell: /bin/bash
userPassword:: e1NTSEF9Nkl0M2c0a2VqUDU0ei8yd00xQjBjUmdKM0lQVzNsN0E=
sn:: ZmFycmVsbaO=
mail: farrell@gmail.com
givenName: farrell
structuralObjectClass: inetOrgPerson
entryUUID: 659dabde-8dc4-1039-8e90-0f564f63cf43
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191028114700Z
gidNumber: 10004
entryCSN: 20191107222257.753871Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191107222257Z
```

```
dn: uid=goban,ou=people,dc=naboo,dc=ally
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
objectClass: person
cn:: Z29iYW4K
uid: goban
homeDirectory: /home/nfs/goban
loginShell: /bin/bash
userPassword:: e1NTSEF9Nkl0M2c0a2VqUDU0ei8yd00xQjBjUmdKM0lQVzNsN0E=
sn:: Z29iYW4K
mail: goban@gmail.com
givenName: goban
structuralObjectClass: inetOrgPerson
entryUUID: 86d7953a-8dc4-1039-8e91-0f564f63cf43
creatorsName: cn=admin,dc=naboo,dc=ally
createTimestamp: 20191028114756Z
uidNumber: 2008
gidNumber: 10004
entryCSN: 20191107222307.337927Z#000000#000#000000
modifiersName: cn=admin,dc=naboo,dc=ally
modifyTimestamp: 20191107222307Z
```

To check if everything is well configured we can use the LDAP Account Manager, This is a web manager for our ldap. We will install it in our https/https machine using this commands:

apt -y install ldap-account-manager

If we try to connect now it will show an error, that's why we have to install this dependency:

apt-get install php-zip

Then we restart the service and check if it works properly

service apache2 restart

service apache2 status

In order to access we will use this link

<http://192.168.40.3/lam>

4 Create a digital certificate and ask ca.coruscant.capital to sign it

Hecho

We must create a digital certificate and ask ca.coruscant.capital to sign it. This digital certificate will be used in all the xxxx.ally network services in which a digital certificate is required.

The first step is to create a key and the Certificate Signing Request using OpenSSL.
Create certificate

In our case we are going to create the key in our Router just for comfort.

Router:

With this command we will create a key and the certificate signing request all at the same time:

```
openssl req -new -newkey rsa:2048 -nodes -keyout naboo.key -out naboo.csr
```

Then we copy the csr file to debian10 machine in coruscant capital.

```
scp naboo.csr si2@172.20.202.252:
```

ca.coruscant.capital

Now we are going to use our csr file and the ca file of coruscant to create a certificate signed by ca.coruscant.capital

Naboo
Julen Fernández
Ibon Pina
SI2

```
openssl ca -config /CA/openssl.cnf -policy policyAnything -out naboo.crt  
-infiles /home/si2/naboo.csr
```

Then we check if the certificate is formed in a pem file or not.

```
root@debian10-si2:/CA/nuevoscerts# cat naboo.crt
Certificate:
Data:
Version: 3 (0x2)
Serial Number: 8 (0x8)
Signature Algorithm: sha256WithRSAEncryption
Issuer: C=RC, ST=Corusca, O=Republic, OU=Republic, CN=ca.coruscant.capital
Validity
Not Before: Nov 6 23:06:41 2019 GMT
Not After : Nov 5 23:06:41 2020 GMT
Subject: C=NB, ST=naboo, L=naboo, O=naboo, OU=naboo, CN=naboo/emailAddress=naboo@gmail.com
Subject Public Key Info:
Public Key Algorithm: rsaEncryption
RSA Public-Key: (2048 bit)
Modulus:
00:bc:d5:63:9b:20:88:86:e3:2a:59:0b:f8:52:8f:
d2:7a:b0:bc:a7:a4:bd:14:84:4e:47:8b:d7:cd:11:
5c:fb:4c:3b:b6:15:03:cd:29:7b:0a:a4:98:36:26:
35:ce:d9:96:82:c0:31:a8:c4:23:15:a9:b1:b0:65:
84:42:50:66:bd:65:56:ee:d4:53:52:34:49:e6:ed:
86:b6:98:c1:42:3c:a9:43:13:24:c8:83:5c:7c:de:
56:fe:5a:b4:cc:5d:b5:3c:c2:f2:b7:91:4c:2c:01:
98:13:51:50:10:fa:07:5e:53:00:09:59:d1:70:e2:
d1:ac:18:bf:45:4b:56:6c:3a:ae:05:19:f5:7a:ee:
58:cc:ee:2f:36:7d:29:8d:ac:0f:c1:56:db:5c:3d:
5e:96:a6:d6:18:7d:e0:be:f1:d3:cb:c8:95:72:23:
fc:46:82:be:de:f4:c7:a1:1f:29:ae:45:49:4f:b1:
8a:ee:b3:e5:91:b8:de:31:02:db:7b:ac:9d:13:a8:
72:76:f4:f4:6c:a1:e1:e1:de:2a:30:24:41:0c:06:
d2:6f:1a:1f:7e:2f:52:56:68:92:2f:c2:e9:6f:a0:
de:ac:3a:93:e2:dc:92:a6:8a:58:df:c7:83:b2:12:
30:b5:82:03:88:e5:92:ab:2d:d9:10:83:40:fd:26:
75:b5
Exponent: 65537 (0x10001)
X509v3 extensions:
X509v3 Basic Constraints:
CA:FALSE
Netscape Comment:
OpenSSL Generated Certificate
X509v3 Subject Key Identifier:
CC:F2:BB:58:22:E4:2A:EF:00:30:49:31:4A:95:FB:4F:4C:D4:FD:CB
X509v3 Authority Key Identifier:
keyid:20:5B:70:89:72:1C:79:07:A5:5E:2A:4A:7A:7B:0B:78:8C:1B:BB:B9

Signature Algorithm: sha256WithRSAEncryption
9e:0d:7c:ac:54:99:b0:67:16:69:39:2d:31:c8:8f:0d:94:3a:
a9:28:55:64:38:36:46:7c:a4:fc:6d:77:4f:fd:d1:23:06:64:
03:2e:da:05:2e:38:52:6b:18:5e:3a:74:34:cd:cd:4a:d1:9a:
d8:5d:62:4c:07:dd:95:38:d4:9c:2b:50:10:38:0d:f9:6b:76:
3a:00:ba:72:7f:c9:90:1d:9a:d1:17:41:1d:39:f4:55:e2:fa:
73:02:51:55:d7:d3:8e:21:23:43:45:09:76:c9:e8:f8:08:d6:
89:ca:4c:ec:9b:fb:60:63:74:2c:77:36:d6:f3:32:8f:36:87:
bd:e1:52:ad:49:76:c3:94:fb:ee:68:d4:b3:75:fe:d7:dd:75:
5b:63:6e:c2:d3:17:c4:39:48:1e:97:54:57:ce:9b:3d:f6:85:
a0:04:1c:09:a5:8e:31:90:9c:2e:4d:66:f6:c9:5e:7f:e5:e2:
3a:36:7a:88:98:51:84:72:85:ce:06:45:2b:06:ba:8c:b8:16:
5a:fb:17:22:b9:62:e1:6c:3f:5c:54:10:8c:41:25:6a:23:90:
```

Naboo
Julen Fernández
Ibon Pina
SI2

```

9c:ee:2a:8e:77:7a:15:ad:ee:26:0e:11:ee:6f:b8:19:85:54:
51:86:7e:48:5d:8d:97:d2:56:c7:c7:cb:8a:63:ab:e8:44:5c:
8e:45:53:19
----BEGIN CERTIFICATE----
MIID1zCCAr+gAwIBAgIBCDANBgkqhkiG9w0BAQsFADBkMQswCQYDVQQGEwJSQzEQ
MA4GA1UECAwHQ29ydXNjYTERMA8GA1UECgwUmVwdWJsaWMxETAPBgNVBAsMCFJI
cHvibGljMR0wGwYDVQQDDBRjYS5jb3J1c2NhbQuY2FwaXRhbDAeFw0xOTExDMDYy
MzA2NDFaFw0yMDExMDUyMzA2NDFaMH0xCzAJBgNVBAYTAk5CMQ4wDAYDVQQIDAVu
YWJvbzEOMAwGA1UEBwwFbmFib28xDjAMBgNVBAoMBW5hYm9vMQ4wDAYDVQLDAVu
YWJvbzEOMAwGA1UEAwwFbmFib28xHjAcBgkqhkiG9w0BCQEWD25hYm9vQGdtYWls
LmNbTCCASlwDQYJKoZlhcNAQEBBQADggEPADCCAQoCggEBALzVY5sgilbjKkl
+FKP0nqwvKekvRSETkeL180RXPtMO7YVA80pewqkmDYmNc7ZloLAMajElxWpsbBl
hEJQZr1Vv7UU10SebthraYwUl8qUMTJMIDXHzeVv5atMxdTzC8reRTCwBmBNR
UBD6B15TAAZ0XDioawYv0VLVm6rgUZ9XruWMzuLzZ9KY2sD8FW21w9Xpam1hh9
4L7x08vIIXlj/EaCvt70x6EfKa5FSU+xiu6z5ZG43jEC23usnROocnb09Gyh4eHe
KjAkQQwG0m8aH34vUIZoki/C6W+g3qw6k+LckqaKWN/Hg7ISMLWCA4jlkqst2RCD
QP0mdbUCAwEEAaN7MHkwCQYDVVR0TBAlwADAsBglghkgBhvhCAQ0EHxYdT3BbINT
TCBHZW5lcmF0ZWQgQ2VydGlmaWNhdGUwHQYDVVR0OBYYEFMzyu1gi5CrwADBJMUqv
+09M1P3LMB8GA1UdlwQYMBaAFCCbcllyHHkHpV4qSnp7C3iMG7u5MA0GCSqGSib3
DQEBCwUAA4IBAQCeDXysVJmwZxZpOS0xyI8NIDqpKFVkJODZGfKT8bXdP/dEjBmQD
LtoFLjhSaxheOnQ0zc1K0zYXWJMB92VONScK1AQOA35a3Y6ALpyf8mQHZrRF0Ed
OfRV4vpzAlFV19OOISNDRQI2yej4CNaJykzsm/tgY3QsdzbW8zKPNoe94VKtSXbD
IPvuaNSzdf7X3XVbY27C0xfEOUgelRXzps99oWgBBwJpY4xkJwuTWb2yV5/5el6
NnqlmFGEcoXOBkUrBrqMuBZa+xciuWLhbD9cVBCMQSvq15Cc7iqOd3oVre4mDhHu
b7gZhVRRhn5IXY2X0lbHx8uKY6voRFyORVMZ
----END CERTIFICATE----

```

In this case our certificate is not in a pem format, as it shows information above the encrypted certificate.

Then we will send it to a machine of our network, in this case to the router.

```
scp naboo.crt debian-router@172.20.202.34:
```

And after this we will send it to our mail server machine, were we will use it for our irc and mail services.

```
scp naboo.crt si2@192.168.40.4:
```

IRC server

In order to convert our crt file into a pem file we can use this command.

```
openssl x509 -in naboo.crt -out naboo.pem -outform PEM
```

If we do “cat naboo.pem” this is the new output:

```

root@debian-10-4:/etc/inspircd/sslcerts# cat naboo.pem
----BEGIN CERTIFICATE----
MIID1zCCAr+gAwIBAgIBCDANBgkqhkiG9w0BAQsFADBkMQswCQYDVQQGEwJSQzEQ
MA4GA1UECAwHQ29ydXNjYTERMA8GA1UECgwUmVwdWJsaWMxETAPBgNVBAsMCFJI
cHvibGljMR0wGwYDVQQDDBRjYS5jb3J1c2NhbQuY2FwaXRhbDAeFw0xOTExDMDYy
MzA2NDFaFw0yMDExMDUyMzA2NDFaMH0xCzAJBgNVBAYTAk5CMQ4wDAYDVQQIDAVu
YWJvbzEOMAwGA1UEBwwFbmFib28xDjAMBgNVBAoMBW5hYm9vMQ4wDAYDVQLDAVu
YWJvbzEOMAwGA1UEAwwFbmFib28xHjAcBgkqhkiG9w0BCQEWD25hYm9vQGdtYWls
LmNbTCCASlwDQYJKoZlhcNAQEBBQADggEPADCCAQoCggEBALzVY5sgilbjKkl
+FKP0nqwvKekvRSETkeL180RXPtMO7YVA80pewqkmDYmNc7ZloLAMajElxWpsbBl
hEJQZr1Vv7UU10SebthraYwUl8qUMTJMIDXHzeVv5atMxdTzC8reRTCwBmBNR
UBD6B15TAAZ0XDioawYv0VLVm6rgUZ9XruWMzuLzZ9KY2sD8FW21w9Xpam1hh9
4L7x08vIIXlj/EaCvt70x6EfKa5FSU+xiu6z5ZG43jEC23usnROocnb09Gyh4eHe
KjAkQQwG0m8aH34vUIZoki/C6W+g3qw6k+LckqaKWN/Hg7ISMLWCA4jlkqst2RCD
QP0mdbUCAwEEAaN7MHkwCQYDVVR0TBAlwADAsBglghkgBhvhCAQ0EHxYdT3BbINT
TCBHZW5lcmF0ZWQgQ2VydGlmaWNhdGUwHQYDVROOBYYEFMzyu1gi5CrwADBJMUqv

```

Naboo

Julen Fernández

Ibon Pina

SI2

```
+09M1P3LMB8GA1UdlwQYMBaAFCBbcclyHHkHpV4qSnp7C3iMG7u5MA0GCSqGSIB3
DQEBCwUA4IBAQCeDXysVJmwZxZpOS0xyI8NIDqpKFVkJODZGfKT8bXdP/dEjBmQD
LtoFLjhSaxheOnQ0zc1K0ZrYXWJMB92VONSCK1AQOA35a3Y6ALpyf8mQHZrRF0Ed
OfRV4vpzAlFV19O0ISNDRQI2yej4CNaJykzsm/tgY3QsdzbW8zKPNoe94VKtSXbD
IPvuaNSzdf7X3XVbY27C0xfEOUgel1RXzps99oWgBBwJpY4xkJwuTWb2yV5/5eI6
NnqlmFGEcoXOBkUrBrqMuBza+xciuWLhbD9cVBCMQSvql5Cc7iqOd3oVre4mDhHu
b7gZhVRRhn5lXY2X0lbHx8uKY6voRFyORVMZ
-----END CERTIFICATE-----
```

5 Install and configure an electronic email server in naboo.ally

Meter al dominio

we will install and configure an electronic mail server in xxxx.ally network (postfix, dovecot and squirrelmail), adding the digital certificate signed by ca.coruscant.capital. Each user will have its own electronic mail account (the LDAP server will be used for user's login), will receive in the same inbox the electronic mails addressed to user@xxxx.ally and user@xxxx.local, user groups with more than one user will work as distribution groups (as, for example, for sending an email to all the pilots using only one address), user groups with only one user will work as aliases (an additional name), and each user will have an inbox of 30 MB, being the message maximum size 1 MB, not to overload the network nor the servers.

6 Install and configure an instant messaging server in xxxx.ally

work in progress - ibon

we will install and configure an instant messaging server in xxxx.ally network (inspircd), adding the digital certificate signed by ca.coruscant.capital. Each user can access the instant messaging server with a login against the LDAP server. Server administrator will be Yoda. Service operators will be users of groups Jedi Masters and Generals, which will be able to connect from networks xxxx.ally and xxxx.local. The rest of the users will be able to connect only from xxxx.local. Connections from 172.20.202.0/24 or 172.20.14.0/24 will not be allowed. There cannot be more than 10 users connected at the same time and the next nicknames will be forbidden: Vader and DarthVader

In order to communicate with each other in our network we will use an irc server, this server will be hosted in the same machine as the mail server, the one with the ip 192.168.40.4. This irc will only allow users logged against the ldap, and will also ban users with nicknames like Vader or DarthVader.

The irc service that we are going to use is called inspircd, is a simple easy to configure irc that allows use to use some modules such as ldap authentication and ssl verifying.

The first step is to install inspircd, for this we will use the command:

Naboo
Julen Fernández
Ibon Pina
SI2

apt-get install inspircd

This command will install the version 2 of inspircd.

Now is time for the configuration, we used a template so that we can just tweak little details in the config file instead of writing it all.

The config file is stored in /etc/inspircd/inspircd.conf

This is each configuration explained

1- This is where we specify the domain name of our irc server and the description that will be shown when establishing connection.

```
<server name="irc.naboo.ally"
        description="Naboo IRC Server"
        network="naboo.ally"
        id="46T">
```

2- Admin name, nickname and email.

```
<admin name="yoda"
      nick="yoda"
      email="yoda@gmail.com">
```

3- This is where we establish both shutdown and restart passwords.

```
<power diepass="M@ythe4th" restartpass="M@ythe4th" pause="2">
```

4- These three entries are the operators in our irc. carlissian, yoda and kenobi.

```
<oper name="carlissian"
      password="M@ythe4th"
      host="*@192.168.*.*"
      type="NetAdmin">

<oper name="yoda"
      password="M@ythe4th"
      host="*@192.168.*.*"
      type="NetAdmin">

<oper name="kenobi"
      password="M@ythe4th"
      host="*@192.168.*.*"
      type="NetAdmin">
```

5- Here we specify the max users and opers in each channel, also the dns server and the location of the pid file.

```
<channels users="20"
          opers="60">

<dns server="192.168.40.2" timeout="5">

<pid file="/var/run/inspircd.pid">
```

6- Here we can find the general options and the security options, we haven't changed anything here.

```

<options prefixquit="Quit: "
    noservices="no"
    qaprefixes="no"
    deprotectself="no"
    deprotectothers="no"
    flatlinks="no"
    hideulines="no"
    syntaxhints="no"
    cyclehosts="yes"
    ircumsgprefix="no"
    announcets="yes"
    disablehmac="no"
    hostintopic="yes"
    quietbursts="yes"
    pingwarning="15"
    allowhalfop="yes"
    exemptchanops="">

<security hidewhois=""
    userstats="Pu"
    customversion=""
    hidesplits="no"
    hidebans="no"
    operspywhois="no"
    hidemodes="eI"
    maxtargets="20">

```

7- In this entry we specify the banned nicknames, such as Vader or DarthVader

```

<badnick nick="ChanServ" reason="Reserved For Services">
<badnick nick="NickServ" reason="Reserved For Services">
<badnick nick="OperServ" reason="Reserved For Services">
<badnick nick="MemoServ" reason="Reserved For Services">
<badnick nick="Vader" reason="Banned User">
<badnick nick="DarthVader" reason="Banned User">

```

In order to add modules we just have to enter them into the config file.

The first module that we are gonna add is the ldap authentication module.

The correct syntax is this one:

```

<module name="m_ldapauth.so">
<ldapauth baserdn="ou=people,dc=naboo,dc=ally"
    attribute="uid"
    server="ldap://192.168.40.2"
    killreason="LDAP auth failed"
    searchscope="subtree"
    binddn="cn=admin,dc=naboo,dc=ally"
    bindauth="M@ythe4th"
    verbose="yes"
    dbid="ldap-users"
    userfield="no">

```

In the module name we specify the name, in this case "m_ldapauth.so"

The next parameter is the baserdn, where we just put the ou "people" created previously in open ldap and the domain.

Then we specify the ip of our ldap server

Another important parameter is binddn, where we just type the data of the administrator of our ldap.

In the bindauth parameter we just enter the password of our ldap.

It is important to use the /connect command properly when using the irssi client chat application, for example, if we are using the ldap auth module we will have to use irssi as a user of the domain, but also we will need to type the password in the “/connect” command, something like this:

/connect “ip-domain” “port” “password”

Once we have configured our inspircd config file we have to restart the service

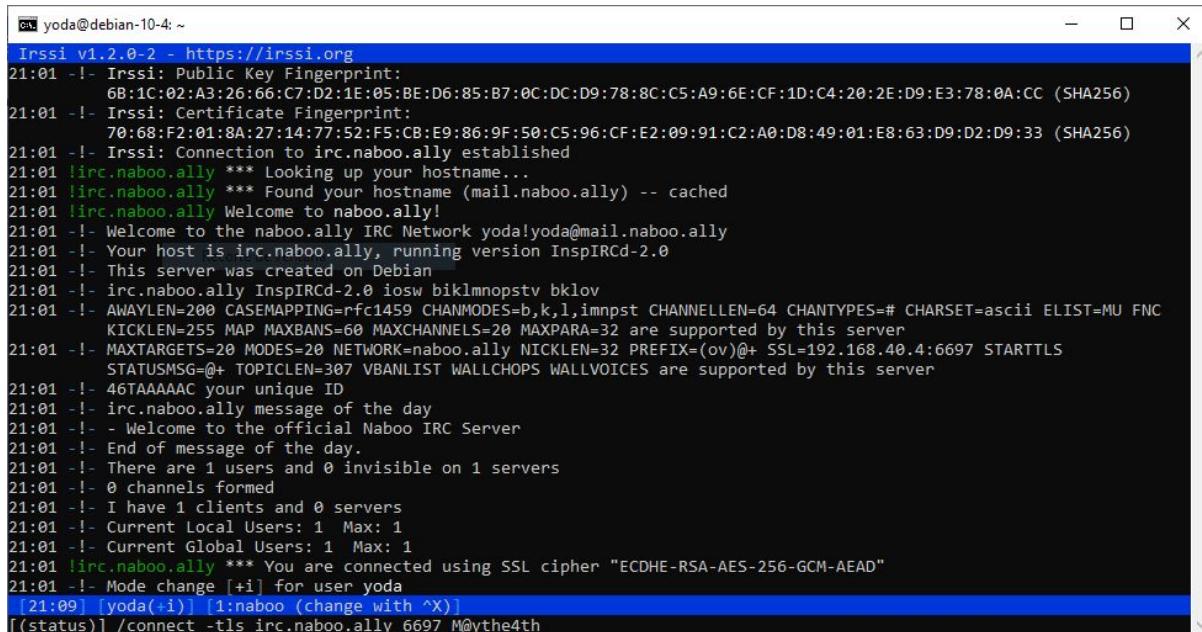
For this task we can use the command:

service inspircd restart

And then we can check the status with this command:

service inspircd status

If all of this works properly we should be only able to connect to the irc using a logged against the ldap user like yoda or carlissian. If not the server should deny the access.



```
yoda@debian-10-4: ~
Irssi v1.2.0-2 - https://irssi.org
21:01 -!- Irssi: Public Key Fingerprint:
          68:1C:02:A3:26:66:C7:D2:1E:05:BE:D6:85:B7:0C:DC:D9:78:8C:C5:A9:6E:CF:1D:C4:20:2E:D9:E3:78:0A:CC (SHA256)
21:01 -!- Irssi: Certificate Fingerprint:
          70:68:F2:01:8A:27:14:75:52:F5:CB:E9:86:9F:50:C5:96:CF:E2:09:91:C2:A0:D8:49:01:E8:63:D9:D2:D9:33 (SHA256)
21:01 -!- Irssi: Connection to irc.naboo.ally established
21:01 !irc.naboo.ally *** Looking up your hostname...
21:01 !irc.naboo.ally *** Found your hostname (mail.naboo.ally) -- cached
21:01 !irc.naboo.ally Welcome to naboo.ally!
21:01 -!- Welcome to the naboo.ally IRC Network yoda!yoda@mail.naboo.ally
21:01 -!- Your host is irc.naboo.ally, running version InspIRCd-2.0
21:01 -!- This server was created on Debian
21:01 -!- irc.naboo.ally InspIRCd-2.0 iosw biklmnopstv bklov
21:01 -!- AWAYLEN=200 CASEMAPPING=rfc1459 CHANMODES=b,k,l,immpst CHANNELLEN=64 CHANTYPES=# CHARSET=ascii ELIST=MU FNC
          KICKLEN=255 MAP MAXBANS=60 MAXCHANNELS=20 MAXPARA=32 are supported by this server
21:01 -!- MAXTARGETS=20 MODES=20 NETWORK=naboo.ally NICKLEN=32 PREFIX=(ov)@+ SSL=192.168.40.4:6697 STARTTLS
          STATUSMSG:@+ TOPICLEN=307 VBANLIST WALLCHOPS WALLVOICES are supported by this server
21:01 -!- 46TAAAAC your unique ID
21:01 -!- irc.naboo.ally message of the day
21:01 -!- - Welcome to the official Naboo IRC Server
21:01 -!- End of message of the day.
21:01 -!- There are 1 users and 0 invisible on 1 servers
21:01 -!- 0 channels formed
21:01 -!- I have 1 clients and 0 servers
21:01 -!- Current Local Users: 1 Max: 1
21:01 -!- Current Global Users: 1 Max: 1
21:01 !irc.naboo.ally *** You are connected using SSL cipher "ECDHE-RSA-AES-256-GCM-AEAD"
21:01 -!- Mode change [+i] for user yoda
[21:09] [yoda(+i)] [1:naboo (change with ^X)]
[status] /connect -tls irc.naboo.ally 6697 M@ythe4th
```

The second module that we are going to use is called m_ssl_gntuls.so.

With this module we will add the certificates generated before to our irc service.

The most important parameters are the cafile, certfile and keyfile, because those are going to be the routes to the certificates and keys.

The “cafle” is the certificate taken from ca.coruscant

The keyfile is just the key that we used to generate the csr and the crt in previous documentation.

We will need to change the extension of the key from .key to .pem. but in this case we can do it without any problem.

The certfile is the certificate generated with ca.coruscant’s ca file and our csr.

We will also need to convert this .crt file into .pem, which is explained in the documentation of the certificates.

```
<module name="m_ssl_gnutls.so">
<bind address="" port="6697" type="clients" ssl="gnutls">
<gnutls
    cafile="/etc/inspircd/sslcerts2/ca.pem"
    certfile="/etc/inspircd/sslcerts/naboo.pem"
    keyfile="/etc/inspircd/sslcerts/key.pem"
    priority="SECURE192"
    advertisedports="192.168.40.4:6697"
    dhbits="2048"
    hash="sha1"
    showports="yes"
    starttls="yes">
```

```
● inspircd.service - IRC server
   Loaded: loaded (/lib/systemd/system/inspircd.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2019-11-13 07:42:39 CET; 1h 46min ago
     Docs: man:inspircd(1)
 Main PID: 380 (inspircd)
   Tasks: 1 (limit: 1150)
  Memory: 7.0M
    CGroup: /system.slice/inspircd.service
           └─380 /usr/sbin/inspircd --logfile /var/log/inspircd.log --config /etc/inspircd/inspircd.conf --nofork

nov 13 07:42:57 debian-10-4 inspircd[380]: (C) InspIRCd Development Team.
nov 13 07:42:57 debian-10-4 inspircd[380]: Developers:
nov 13 07:42:57 debian-10-4 inspircd[380]:          Brain, FrostyCoolSlug, w00t, Om, Special, peavey
nov 13 07:42:57 debian-10-4 inspircd[380]:          aquanight, psychon, dz, danieldg, jackmcbar
nov 13 07:42:57 debian-10-4 inspircd[380]:          Attila
nov 13 07:42:57 debian-10-4 inspircd[380]: Others: See /INFO Output
nov 13 07:43:00 debian-10-4 inspircd[380]: Loading core commands.....
nov 13 07:43:00 debian-10-4 inspircd[380]: [*] Loading module: m_ssl_gnutls.so
nov 13 07:43:00 debian-10-4 inspircd[380]: [*] Loading module: m_ldapauth.so
nov 13 07:43:01 debian-10-4 inspircd[380]: InspIRCd is now running as 'irc.naboo.ally'[46T] with 1024 max open sockets
~
```

If we want to connect now we will need to use the argument -tls to connect using ssl.

7 Install two network printers and install and configure two printer servers in xxxx.local network that will manage these two printers.

DONETE

install two network printers (one of them is actually a network printer, while the other is local printer we will convert into a network printer using a Raspberry Pi), and install and configure two printer servers in xxxx.local network that will manage these two printers. Pilots group users will not have permission to print anything, while the users on the other three groups (Captains, Generals and Jedi Masters) will have permission. But only Jedi Masters will be able to delete tasks that has been sent to printer.

<https://www.rosehosting.com/blog/how-to-generate-a-self-signed-ssl-certificate-on-linux/>

Naboo
Julen Fernández
Ibon Pina
SI2

[https://www.digitalocean.com/community/tutorials/how-to-set-up-an-irc-server-on-ubuntu-14-04-with-inspircd-2-0-and-shalture#step-6-%E2%80%94-setting-up-ssl-\(optional\)](https://www.digitalocean.com/community/tutorials/how-to-set-up-an-irc-server-on-ubuntu-14-04-with-inspircd-2-0-and-shalture#step-6-%E2%80%94-setting-up-ssl-(optional))

```
apt-get install avahi-daemon avahi-discover libnss-mdns cups cups-pdf python-cups  
200 usermod -aG lpadmin pi  
201 /etc/init.d/cups start  
202 /etc/init.d/avahi-daemon start  
203 ls  
204 cd ..  
205 cd ..  
206 nano /etc/cups/cupsd.conf  
207 /etc/init.d/cups Restart
```

This is the whole config file of cups:

cat /etc/cups/cupsd.conf

```
#  
# Configuration file for the CUPS scheduler. See "man cupsd.conf" for a  
# complete description of this file.  
  
#  
  
# Log general information in error_log - change "warn" to "debug"  
# for troubleshooting...  
LogLevel warn  
PageLogFormat  
  
# Deactivate CUPS' internal logrotating, as we provide a better one, especially  
# LogLevel debug2 gets usable now  
MaxLogSize 0  
  
# Only listen for connections from the local machine.  
#Listen localhost:631  
Port 631  
Listen /run/cups/cups.sock  
  
# Show shared printers on the local network.  
Browsing On  
BrowseLocalProtocols dnssd  
  
# Default authentication type, when authentication is required...  
DefaultAuthType Basic  
  
# Web interface setting...  
WebInterface Yes
```

The main parameters that i have modified are just those where it says <location> where i set “Allow all” in order to connect to the cups web server from anywhere.

```
# Restrict access to the server...  
<Location />  
Allow all  
</Location>  
  
# Restrict access to the admin pages...  
<Location /admin>  
Allow all  
</Location>  
  
# Restrict access to configuration files...  
<Location /admin/conf>  
AuthType Default  
Require user @SYSTEM  
Allow all
```

Naboo
Julen Fernández
Ibon Pina
SI2

```

</Location>

# Restrict access to log files...
Allow, Deny, and Allow @ Local
<Location /admin/log>
AuthType Default
Require user @SYSTEM
Allow all
# Order allow,deny
</Location>

# Set the default printer/job policies...
<Policy default>
# Job/subscription privacy...
JobPrivateAccess default
JobPrivateValues default
SubscriptionPrivateAccess default
SubscriptionPrivateValues default

# Job-related operations must be done by the owner or an administrator...
<Limit Create-Job Print-Job Print-URI Validate-Job>
Order deny,allow
</Limit>

<Limit Send-Document Send-URI Hold-Job Release-Job Restart-Job Purge-Jobs Set-Job-Attributes Create-Job-Subscription
Renew-Subscription Cancel-Subscription Get-Notifications Reprocess-Job Cancel-Current-Job Suspend-Current-Job Resume-Job
Cancel-My-Jobs Close-Job CUPS-Move-Job CUPS-Get-Document>
Require user @OWNER @SYSTEM
Order deny,allow
</Limit>

# All administration operations require an administrator to authenticate...
<Limit CUPS-Add-Modify-Printer CUPS-Delete-Printer CUPS-Add-Modify-Class CUPS-Delete-Class CUPS-Set-Default CUPS-Get-Devices>
AuthType Default
Require user @SYSTEM
Order deny,allow
</Limit>

# All printer operations require a printer operator to authenticate...
<Limit Pause-Printer Resume-Printer Enable-Printer Disable-Printer Pause-Printer-After-Current-Job Hold-New-Jobs Release-Held-New-Jobs
Deactivate-Printer Activate-Printer Restart-Printer Shutdown-Printer Startup-Printer Promote-Job Schedule-Job-After Cancel-Jobs
CUPS-Accept-Jobs CUPS-Reject-Jobs>
AuthType Default
Require user @SYSTEM
Order deny,allow
</Limit>

# Only the owner or an administrator can cancel or authenticate a job...
<Limit Cancel-Job CUPS-Authenticate-Job>
Require user @OWNER @SYSTEM
Order deny,allow
</Limit>

<Limit All>
Order deny,allow
</Limit>
</Policy>

# Set the authenticated printer/job policies...
<Policy authenticated>
# Job/subscription privacy...
JobPrivateAccess default
JobPrivateValues default
SubscriptionPrivateAccess default
SubscriptionPrivateValues default

# Job-related operations must be done by the owner or an administrator...
<Limit Create-Job Print-Job Print-URI Validate-Job>
AuthType Default
Order deny,allow
</Limit>

```

Naboo
 Julen Fernández
 Ibon Pina
 SI2

```

<Limit Send-Document Send-URI Hold-Job Release-Job Restart-Job Purge-Jobs Set-Job-Attributes Create-Job-Subscription
Renew-Subscription Cancel-Subscription Get-Notifications Reprocess-Job Cancel-Current-Job Suspend-Current-Job Resume-Job
Cancel-My-Jobs Close-Job CUPS-Move-Job CUPS-Get-Document>
AuthType Default
Require user @OWNER @SYSTEM
Order deny,allow
</Limit>

# All administration operations require an administrator to authenticate...
<Limit CUPS-Add-Modify-Printer CUPS-Delete-Printer CUPS-Add-Modify-Class CUPS-Delete-Class CUPS-Set-Default>
AuthType Default
Require user @SYSTEM
Order deny,allow
</Limit>

# All printer operations require a printer operator to authenticate...
<Limit Pause-Printer Resume-Printer Enable-Printer Disable-Printer Pause-Printer-After-Current-Job Hold-New-Jobs Release-Held-New-Jobs
Deactivate-Printer Activate-Printer Restart-Printer Shutdown-Printer Startup-Printer Promote-Job Schedule-Job-After Cancel-Jobs
CUPS-Accept-Jobs CUPS-Reject-Jobs>
AuthType Default
Require user @SYSTEM
Order deny,allow
</Limit>

# Only the owner or an administrator can cancel or authenticate a job...
<Limit Cancel-Job CUPS-Authenticate-Job>
AuthType Default
Require user @OWNER @SYSTEM
Order deny,allow
</Limit>

<Limit All>
Order deny,allow
</Limit>
</Policy>

# Set the kerberized printer/job policies...
<Policy kerberos>
# Job/subscription privacy...
JobPrivateAccess default
JobPrivateValues default
SubscriptionPrivateAccess default
SubscriptionPrivateValues default

# Job-related operations must be done by the owner or an administrator...
<Limit Create-Job Print-Job Print-URI Validate-Job>
AuthType Negotiate
Order deny,allow
</Limit>

<Limit Send-Document Send-URI Hold-Job Release-Job Restart-Job Purge-Jobs Set-Job-Attributes Create-Job-Subscription
Renew-Subscription Cancel-Subscription Get-Notifications Reprocess-Job Cancel-Current-Job Suspend-Current-Job Resume-Job
Cancel-My-Jobs Close-Job CUPS-Move-Job CUPS-Get-Document>
AuthType Negotiate
Require user @OWNER @SYSTEM
Order deny,allow
</Limit>

# All administration operations require an administrator to authenticate...
<Limit CUPS-Add-Modify-Printer CUPS-Delete-Printer CUPS-Add-Modify-Class CUPS-Delete-Class CUPS-Set-Default>
AuthType Default
Require user @SYSTEM
Order deny,allow
</Limit>

# All printer operations require a printer operator to authenticate...
<Limit Pause-Printer Resume-Printer Enable-Printer Disable-Printer Pause-Printer-After-Current-Job Hold-New-Jobs Release-Held-New-Jobs
Deactivate-Printer Activate-Printer Restart-Printer Shutdown-Printer Startup-Printer Promote-Job Schedule-Job-After Cancel-Jobs
CUPS-Accept-Jobs CUPS-Reject-Jobs>
AuthType Default

```

Naboo
 Julen Fernández
 Ibon Pina
 SI2

```

Require user @SYSTEM
Order deny,allow
</Limit>

# Only the owner or an administrator can cancel or authenticate a job...
<Limit Cancel-Job CUPS-Authenticate-Job>
AuthType Negotiate
Require user @OWNER @SYSTEM
Order deny,allow
</Limit>

<Limit All>
Order deny,allow
</Limit>
</Policy>

```

Now we can connect access to the administration of our printer via web browser, using our ip and the cups ports: 192.168.4.91:631

CUPS.org Inicio Administración Clases Ayuda Trabajos Impresoras

CUPS 2.2.10

CUPS es el sistema de impresión de código abierto basado en estándares desarrollado por Apple Inc. para macOS® y otros sistemas operativos tipo UNIX®.

CUPS para usuarios

[Descripción de CUPS](#)
[Impresión desde la línea de comandos y opciones](#)
[Foro de usuarios](#)

CUPS para administradores

[Añadir impresoras y clases](#)
[Gestión de políticas de funcionamiento](#)
[Uso de impresoras de red](#)
[Referencia de cupsd.conf](#)

CUPS para desarrolladores

[Introducción a la programación de CUPS](#)
[La API de CUPS](#)
[Programación de filtros y programas de conexión](#)
[Las APIs HTTP e IPP](#)
[Foro de desarrollo](#)

Activar Windows
[Ve a Configuración para activar Windows.](#)

If we want to add a new printer we have to click where it says “Añadir impresoras y clases.”

Then we click on “Añadir impresora”

If our printer is connected via usb, it will appear in the “Impresoras locales” zone.

Añadir impresora

Añadir impresora

- Impresoras locales:** CUPS-PDF (Virtual PDF Printer)
 CUPS-BRF (Virtual Braille BRF Printer)
 Kyocera ECOSYS P2135dn (Kyocera ECOSYS P2135dn)

Impresoras en red descubiertas:

- Otras impresoras en red:** Backend Error Handler
 Equipo o impresora LPD/LPR
 Protocolo de Impresión de Internet IPP (ipp)
 Protocolo de Impresión de Internet IPP (https)
 Protocolo de Impresión de Internet IPP (ipps)
 Protocolo de Impresión de Internet IPP (http)
 AppSocket/HP JetDirect

We just click on it and click on “siguiente”.

Here we can modify the name description and add an ubicacion to the printer.

Añadir impresora

Añadir impresora

Nombre:
(Puede contener cualquier carácter imprimible excepto "/", "#", y espacio)

Descripción:
(Descripción fácilmente leible tal como "HP LaserJet de doble cara")

Ubicación:
(Ubicación fácilmente leible tal como "Lab 1")

Conexión:

Compartición: Compartir esta impresora

Then we will have to select model of the printer, as our model is not listed there we will just select the first one.

Añadir impresora

Añadir impresora

Nombre: Kyocera_ECOSYS_P2135dn
Descripción: Kyocera ECOSYS P2135dn
Ubicación: kyocera-clase-local
Conexión: usb://Kyocera/ECOSYS%20P2135dn?serial=LVL4395601|Kyocera ECOSYS P2135dn
Compartición: compartir esta impresora

Marca: Kyocera

Modelo:
Kyocera CS-250ci - CUPS+Gutenprint v5.3.1 (en)
Kyocera CS-300ci - CUPS+Gutenprint v5.3.1 (en)
Kyocera CS-400ci - CUPS+Gutenprint v5.3.1 (en)
Kyocera CS-500ci - CUPS+Gutenprint v5.3.1 (en)
Kyocera CS-1815 - CUPS+Gutenprint v5.3.1 (en)
Kyocera CS-2525E - CUPS+Gutenprint v5.3.1 (en)
Kyocera CS-3050ci - CUPS+Gutenprint v5.3.1 (en)
Kyocera CS-3225E - CUPS+Gutenprint v5.3.1 (en)
Kyocera CS-3232E - CUPS+Gutenprint v5.3.1 (en)

O proporcione un archivo PPD: Ningún archivo seleccionado

Finally, we can edit the printing options:

CUPS.org Inicio Administración Clases Ayuda Trabajos Impresoras

Cambiar opciones impresora

Establecer opciones predeterminadas de Kyocera_ECOSYS_P2135dn

General Output Control Common Output Control Extra 1 Output Control Extra 2 Output Co

Rótulos Reglas

General

Media Size: A4
Color Model: RGB Color
Color Precision: Normal
Media Source: Standard
Print Quality: Standard
Resolution: Automatic
2-Sided Printing: Off
Shrink Page If Necessary to Fit Borders: Shrink (print the whole page)

We can print a sample page to verify if the printer works properly.

Kyocera_ECOSYS_P2135dn

Kyocera_ECOSYS_P2135dn (inactiva, aceptando trabajos, compartida)

Mantenimiento ▾ Administración ▾

Mantenimiento
Imprimir página de prueba
Pausar impresora
Rechazar trabajos
Mover todos los trabajos
Cancelar todos los trabajos
Opciones predeterminadas: rótulos=none, none papel=iso_a4_210x297mm caras=one-sided

