

CAMPEONATO DE ANDALUCÍA DE FORMACIÓN PROFESIONAL



Proyecto Fin de CFGS: Andalucía Skills 2018

Andrés Ceballos Rodríguez
acrmontellano@gmail.com
Administración de Sistemas Informáticos en Red
IES Triana

Índice

- Visión General	2
- Objetivos del proyecto	2
- Especificaciones	2
- Cisco test	
- Mikrotik test	
- Servicios	
- VLSM	
- Cisco test	5
- Mikrotik test	15
- Servicios	24
- VLSM	54

Visión General

Andalucía Skills es una competición de Formación Profesional a nivel autonómico que enfrenta a distintos alumnos de diferentes centros repartidos por toda Andalucía en su especialidad.

El ganador de esta competición va clasificado para los Spainskills que es una competición a nivel nacional y participa un participante por Comunidad Autónoma.

En mi caso, participé en la modalidad de Administración de Sistemas Informáticos en Red representando a mi centro el IES Triana de Sevilla.

Objetivos del proyecto

Los objetivos de este proyecto son los siguientes:

1. Realización de las diferentes pruebas de Andalucía Skills 2018 realizadas por mi persona en la modalidad de Administración de Sistemas Informáticos en Red.
2. El uso de este proyecto como guía de las pruebas para los alumnos posteriores que vayan a participar en Andalucía Skills.

Especificaciones

La competición en la modalidad de Administración de Sistemas Informáticos en Red consta de cuatro pruebas. Cada prueba con una ponderación distinta. las pruebas consistían en lo siguiente:

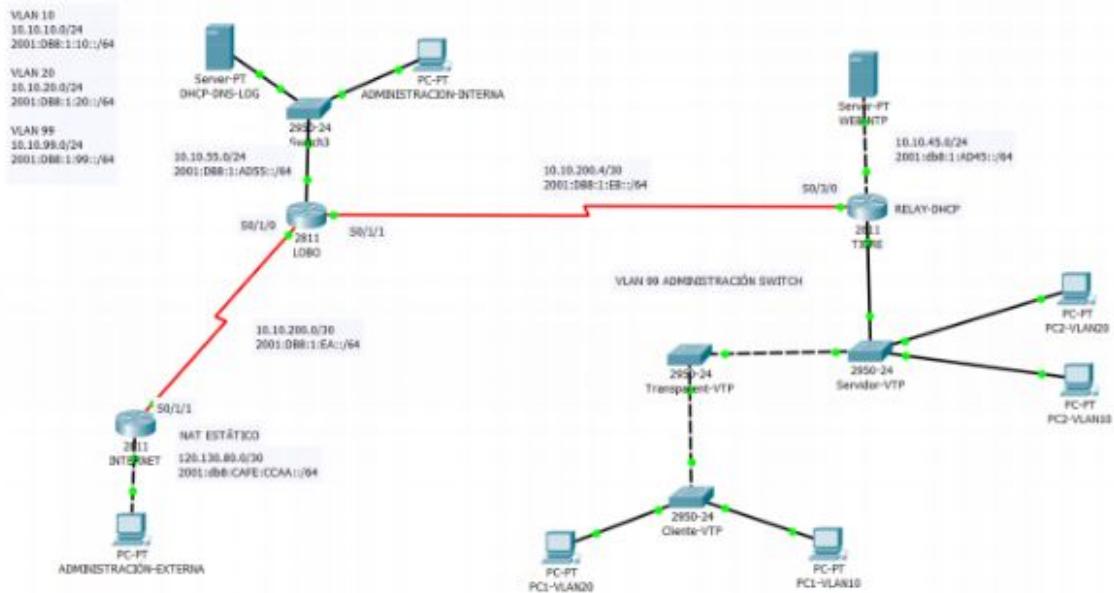
1. Cisco test

La prueba consistía en el montaje de una red simulada usando el programa de Cisco Packet Tracer. En esta prueba, se piden diferentes apartados en el que cada apartado tenía una ponderación a la hora de la corrección.

Esta red, estaba formada por una serie de routers, switches, servidores y equipos que deberían ser configurados tal y como se pide en las pruebas.

Encontramos desde enrutamiento dinámico con protocolo OSPF como servidores DHCP, DHCP Relay, VLAN, configuración de la NAT, ACL's etc.

La red simulada era la siguiente:

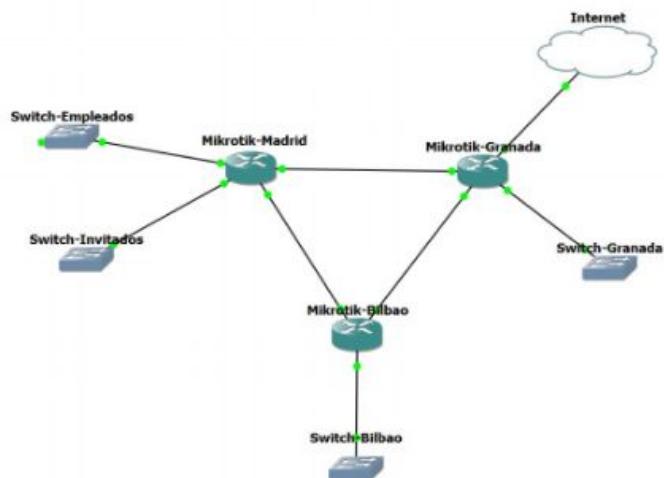


2. Mikrotik test

Esta prueba, consiste en la configuración de Routers Físicos de Mikrotik para su puesta en marcha en una red. Consistía de tres routers físicos que para su configuración debían estar conectados a través de Telnet usando el programa Putty en Windows 10.

Estos routers debían estar conectados entre sí para su correcta configuración y enrutamiento, así como la asignación de IP's correctas por parte del Router que hace como DHCP.

El esquema de red que se pedía en esta prueba era el siguiente:



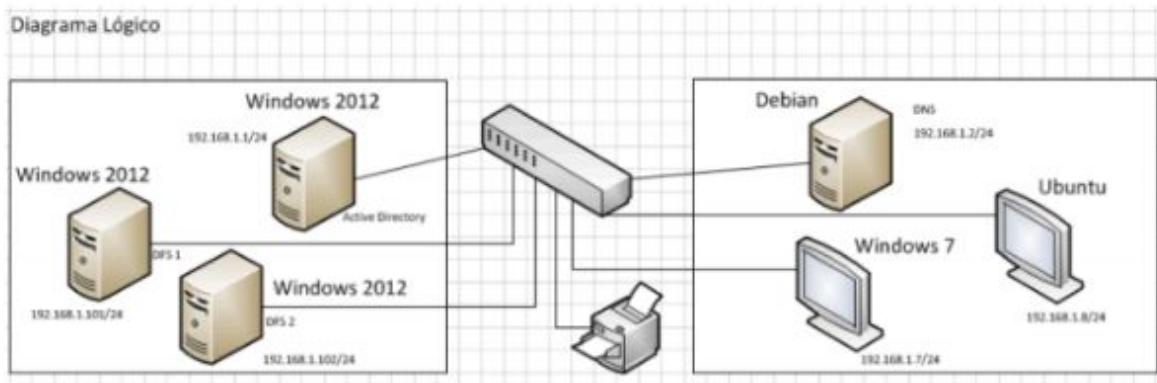
3. Servicios.

En esta prueba, el objetivo consiste en la creación de máquinas virtuales y configuración de distintos Servidores con sus respectivos clientes.

En este caso, se utilizaron máquinas virtuales con Windows Server 2012 y Debian 8 como servidores y como clientes Windows 7 y Ubuntu.

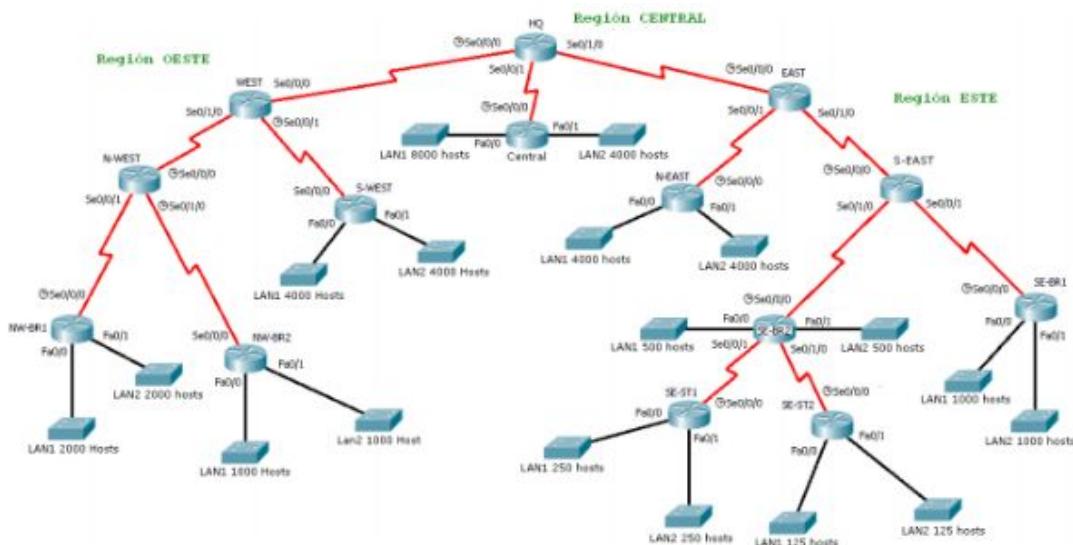
Se realizaron Controladores de Active Directory, Servidores de Ficheros DFS, políticas de Dominio de Windows Server, Servidores Samba para la unión de equipos Ubuntu al Directorio activo de Windows Server.

La estructura era la siguiente:



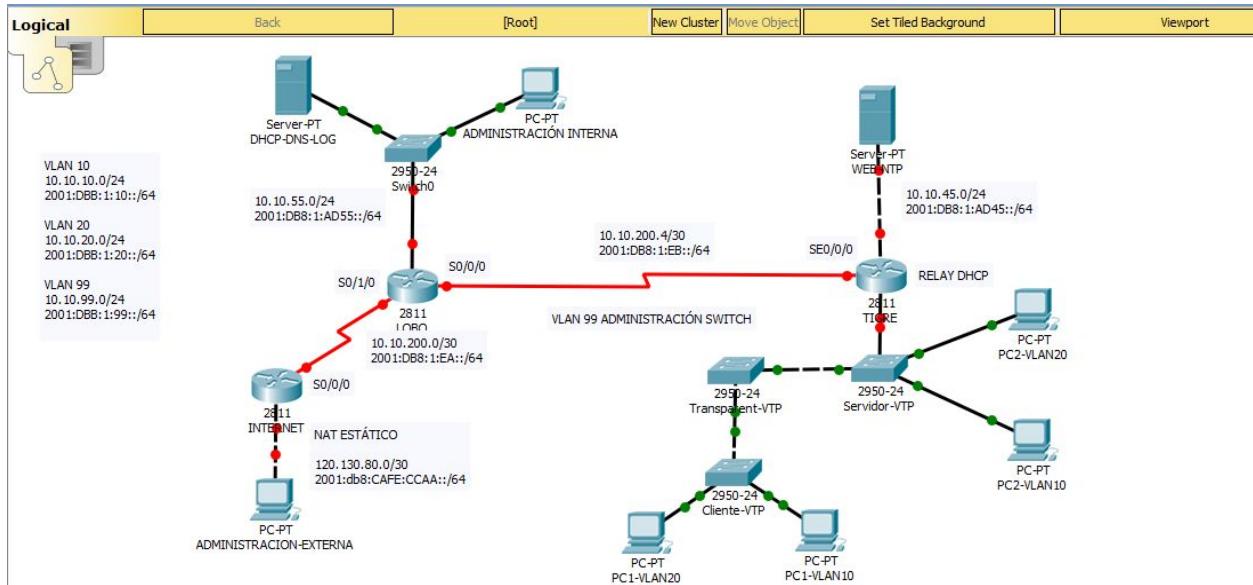
4. VLSM

Esta prueba, consistía en hacer subnetting de la Red 172.16.0.0/16 para aprovechar de forma más eficiente todas las redes del escenario atendiendo al número de host de cada dispositivo para que haya comunicación de extremo a extremo en la red. Para ello, no se debían usar calculadoras IP de ni de otro tipo. El esquema de red era el siguiente:



1. Cisco test

Lo primero que deberemos hacer será dibujar en PacketTracer la red con los nombres de cada equipo y las descripciones que encontramos en las pruebas para mantenerlos organizados.



Una vez realizado el esquema, pasamos a poner nombre a los routers. Para ello, usamos el comando `hostname`.

```
Router>enable
Router#config
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
Router(config)#hostname LOBO
```

Lo siguiente que haremos será darle ip a cada boca de cada router, excepto al router que va directo para el Servidor VTP.

```
LOBO(config-if)#ip a
LOBO(config-if)#ip add
LOBO(config-if)#ip address 10.10.55.1 255.255.255.0
LOBO(config-if)#no sh

LOBO(config-if)#
*LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

*LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0,
changed state to up
```

Para la del router conectado a Servidor VTP, que en este caso es TIGRE, lo que vamos a hacer es configurar subinterfaces.

Lo haremos de la siguiente forma:

```
TIGRE(config)#interface fastEthernet 0/1.1
TIGRE(config-subif)#encapsulation dot1Q 10
TIGRE(config-subif)#ip address 10.10.10.1 255.255.255.0
TIGRE(config-subif)#exit
TIGRE(config)#interface fastEthernet 0/1.2
TIGRE(config-subif)#encapsulation dot1Q 20
TIGRE(config-subif)#ip address 10.10.20.1 255.255.255.0
TIGRE(config-subif)#exit
TIGRE(config)#in
TIGRE(config)#interface fa
TIGRE(config)#interface fastEthernet 0/1
TIGRE(config-if)#no sh

TIGRE(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1,
changed state to up

%LINK-5-CHANGED: Interface FastEthernet0/1.1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface
```

Ahora, pasaremos a configurar los switches.

Primero, cambiaremos el nombre:

```
Switch>enable
Switch#config
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#hostname Servidor-VTP
Servidor-VTP(config)#+
```

Ahora, pasamos a configurar las VLAN.

```
Servidor-VTP(config)#vlan 10
Servidor-VTP(config-vlan)#name VLAN10
Servidor-VTP(config-vlan)#exit
Servidor-VTP(config)#vlan 20
Servidor-VTP(config-vlan)#name VLAN20
```

Ahora, pondremos las bocas en modo trunk.

```
Servidor-VTP(config-if)#switchport mode trunk
Servidor-VTP(config-if)#switchport mode trunk

Servidor-VTP(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/24,
changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/24,
changed state to up

Servidor-VTP(config-if)#sw
Servidor-VTP(config-if)#switchport trunk
Servidor-VTP(config-if)#switchport trunk allow
Servidor-VTP(config-if)#switchport trunk allowed vlan all
```

Ahora, configuramos el switch para ser servidor VTP.

```
Servidor-VTP(config)#vtp domain andaluciaskills
Changing VTP domain name from NULL to andaluciaskills
Servidor-VTP(config)#vtp password andalucia
Setting device VLAN database password to andalucia
Servidor-VTP(config)#vtp mode server
Device mode already VTP SERVER.
Servidor-VTP(config)#+
```

El siguiente paso, será dirigirnos al switch que será transparente.

Para ello, entramos en él y cambiamos el nombre como hicimos anteriormente.

```
Switch>enable
Switch#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#
Switch(config)#hostname Transparent-VTP
Transparent-VTP(config)#+
```

Ahora, pasamos a configurar los enlaces de red en modo trunk:

```
Transparent-VTP(config)#interface fa
Transparent-VTP(config)#interface fastEthernet 0/24
Transparent-VTP(config-if)#swi
Transparent-VTP(config-if)#switchport mode
Transparent-VTP(config-if)#switchport mode trunk
Transparent-VTP(config-if)#switchport mode trunk
Transparent-VTP(config-if)#sw
Transparent-VTP(config-if)#switchport trunk
Transparent-VTP(config-if)#switchport trunk allow
Transparent-VTP(config-if)#switchport trunk allowed vlan all
```

Ahora, con VTP usamos a nuestro switch como transparente, para ello lo haremos con los siguientes comandos:

```
Transparent-VTP(config)#vtp domain andaluciaskills
Domain name already set to andaluciaskills.
Transparent-VTP(config)#vtp password andalucia
Setting device VLAN database password to andalucia
Transparent-VTP(config)#vtp mode transparent
Setting device to VTP TRANSPARENT mode.
```

Ahora, nos quedaría el switch de cliente que sería igual:

```
Switch(config)#vtp domain andaluciaskills
Domain name already set to andaluciaskills.
Switch(config)#vtp password andalucia
Setting device VLAN database password to andalucia
Switch(config)#vtp mode client
Setting device to VTP CLIENT mode.
```

Una vez realizadas estas configuraciones, ya tendremos las VLAN en nuestro cliente dadas por el servidor VTP. Ahora, pondremos los puertos en modo access y le diremos que vlan será la correspondiente a cada puerto:

```
Switch(config)#inte
Switch(config)#interface fa
Switch(config)#interface fastEthernet 0/1
Switch(config-if)#swit
Switch(config-if)#switchport mode
Switch(config-if)#switchport mode a
Switch(config-if)#switchport mode access
Switch(config-if)#sw
Switch(config-if)#switchport a
Switch(config-if)#switchport access vl
Switch(config-if)#switchport access vlan 20
```

Ahora, lo siguiente que vamos a realizar es configurar la vlan administrativa. Para ello, en el switch creamos la vlan administrativa que será la VLAN 99. Para ello, añadimos la vlan primero:

```
Servidor-VTP(config)#vlan 99
Servidor-VTP(config-vlan)#name administrativa
Servidor-VTP(config-vlan)#do wr
Building configuration...
[OK]
```

Lo siguiente que haremos será configurar el switch para habilitarla:

```
Servidor-VTP(config)#interface fastEthernet 0/1
Servidor-VTP(config-if)#sw
Servidor-VTP(config-if)#switchport tru
Servidor-VTP(config-if)#switchport trunk allo
Servidor-VTP(config-if)#switchport trunk na
Servidor-VTP(config-if)#switchport trunk native vla
Servidor-VTP(config-if)#switchport trunk native vlan 99
Servidor-VTP(config-if)#sw
Servidor-VTP(config-if)#switchport mode
Servidor-VTP(config-if)#switchport mode trunk
Servidor-VTP(config-if)#switchport mode trunk
Servidor-VTP(config-if)#in
Servidor-VTP(config-if)#interface vlan99
Servidor-VTP(config-if)#ip address 10.10.99.2 255.255.255.0
Servidor-VTP(config-if)#do wr
```

Ya tendremos la vlan administrativa habilitada. Con esto, tendremos configurados los Switchs. Ahora pasamos a la configuración de los routers.

Lo primero que vamos a hacer, es el protocolo ppp entre Router Tigre y Lobo. Empezaremos con Lobo:

```
LOBO(config)#username TIGRE password andalucia
LOBO(config)#in
LOBO(config)#interface Se
LOBO(config)#interface Serial 0/0/0
LOBO(config-if)#enca
LOBO(config-if)#encapsulation ppp
LOBO(config-if)#
*LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0,
changed state to down

LOBO(config-if)#ppp au
LOBO(config-if)#ppp authentication PAP
LOBO(config-if)#ppp pa
LOBO(config-if)#ppp pap se
LOBO(config-if)#ppp pap sent-username LOBO password andalucia
LOBO(config-if)#do wr
Building configuration...
[OK]
```

Pasamos a Tigre:

```
TIGRE(config)#username LOBO password andalucia
TIGRE(config)#in
TIGRE(config)#interface se
TIGRE(config)#interface serial 0/0/0
TIGRE(config-if)#encap
TIGRE(config-if)#encapsulation ppp
TIGRE(config-if)#pp
TIGRE(config-if)#ppp
TIGRE(config-if)#ppp a
TIGRE(config-if)#ppp authentication PAP
TIGRE(config-if)#ppp pa
TIGRE(config-if)#ppp pap se
TIGRE(config-if)#ppp pap sent-username TIGRE password andalucia
TIGRE(config-if)#do wr
Building configuration...
[OK]
```

Con esto ya tenemos habilitado el protocolo ppp entre estos dos Routers.

Ahora, pasamos a configurar OSPF.

```
LOBO(config-router)#network 10.10.55.0 ?
  A.B.C.D  OSPF wild card bits
LOBO(config-router)#network 10.10.55.0 0.0.0.255 area 1
LOBO(config-router)#network 10.10.200.0 0.0.0.3 area 1
LOBO(config-router)#network 10.10.200.4. 0.0.0.3 area 1
^
* Invalid input detected at '^' marker.

LOBO(config-router)#network 10.10.200.4 0.0.0.3 area 1
```

```

INTERNET>enable
INTERNET#config
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line. End with CNTL/Z.
INTERNET(config)#router os
INTERNET(config)#router ospf 1
INTERNET(config)#router ospf 1
INTERNET(config-router)#ne
INTERNET(config-router)#net
INTERNET(config-router)#network 10.10.200.0 0.0.0.3 area 1
INTERNET(config-router)#do wr
Building configuration...
[OK]
INTERNET(config-router)#exit
INTERNET(config)#
02:40:13: %OSPF-5-ADJCHG: Process 1, Nbr 10.10.200.5 on
Serial0/0/0 from LOADING to FULL, Loading Done

```

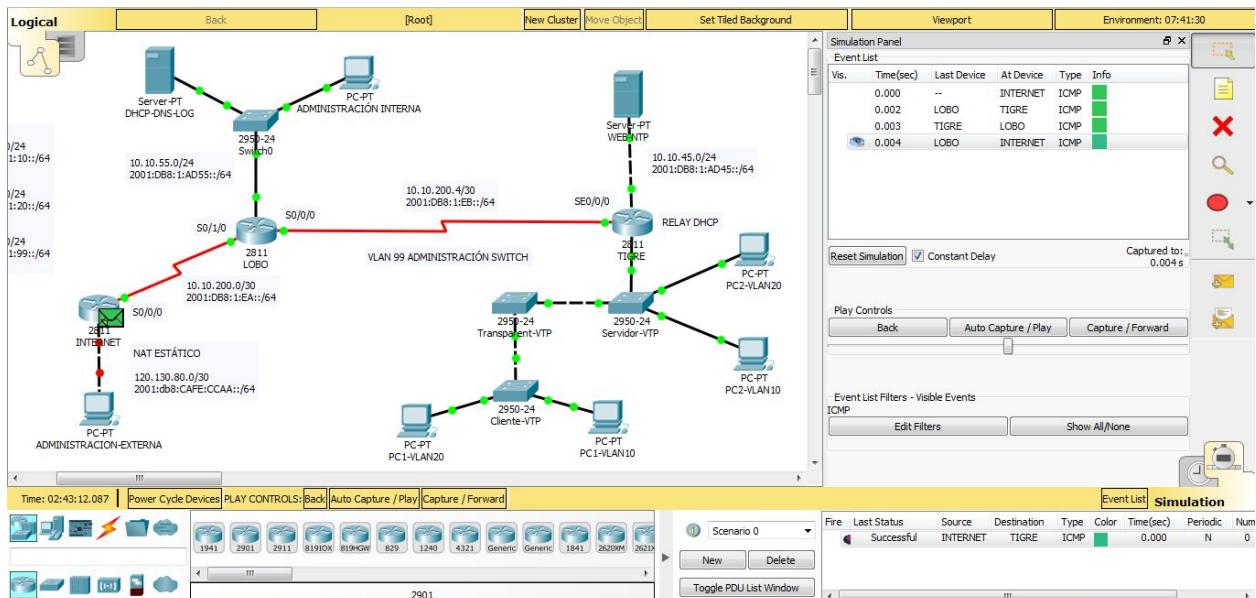
```

TIGRE>enable
TIGRE#
TIGRE#config
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line. End with CNTL/Z.
TIGRE(config)#router ospf 1
TIGRE(config-router)#network 10.10.45.0 0.0.0.255 area 1
TIGRE(config-router)#network 10.10.10.0 0.0.0.255 area 1
TIGRE(config-router)#network 10.10.20.0 0.0.0.255 area 1
TIGRE(config-router)#network 10.10.99.0 0.0.0.255 area 1
TIGRE(config-router)#network 10.10.200.4 0.0.0.3 area 1
TIGRE(config-router)#
02:41:34: %OSPF-5-ADJCHG: Process 1, Nbr 10.10.200.5 on
Serial0/0/0 from LOADING to FULL, Loading Done

```

con esto tendríamos ospf configurado.

Ahora, vamos a probar que funcionan. Enviaremos un paquete ICMP de un router a otro.

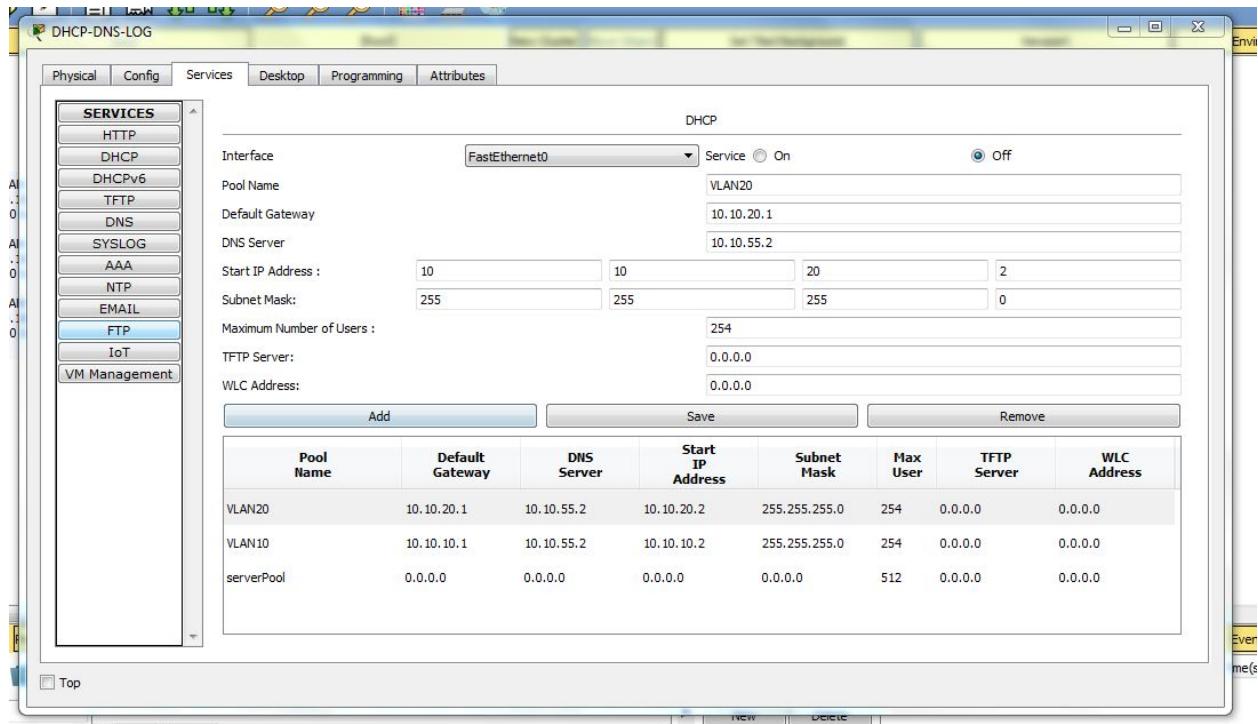


Una vez tengamos funcionando OSPF, lo que vamos a hacer ahora es configurar DHCP Relay y el Servidor DHCP.

El DHCP Relay lo habilitaremos en el router Tigre y LOBO.

Lo primero que vamos a hacer, será configurar el servidor DHCP. Nos dirigimos al Servidor.

En la pestaña de servicios hacemos click en DHCP y lo configuraremos de la siguiente manera:



Ahora, cambiamos la IP a nuestro Servidor y ya lo tendríamos funcionando.

Pasamos a configurar el DHCP Relay:

Para ello, nos dirigiremos a las bocas del router que van a pasar nuestras peticiones DHCP usaremos el comando **ip helper-address x.x.x.x** las x se corresponde con la dirección del Servidor DHCP.

```

TIGRE(config)#interface fa
TIGRE(config)#interface fastEthernet 0/1.1
TIGRE(config-subif)#ip helper-address 10.10.55.2
TIGRE(config-subif)#exit
TIGRE(config)#interface fastEthernet 0/1.2
TIGRE(config-subif)#ip helper-address 10.10.55.2
TIGRE(config-subif)#exit
TIGRE(config)#interface fastEthernet 0/1.2
TIGRE(config-subif)#ip helper-address 10.10.55.2
TIGRE(config-subif)#exit
TIGRE(config)#do wr
Building configuration...
[OK]

```

Ya tenemos configurado nuestro DHCP. Ahora lo que vamos a pasar es a configurar el Servidor NTP y LOG para tener todos los equipos en hora. Para ello, lo habilitamos desde el servidor que vayamos a utilizar como servidor ntp.

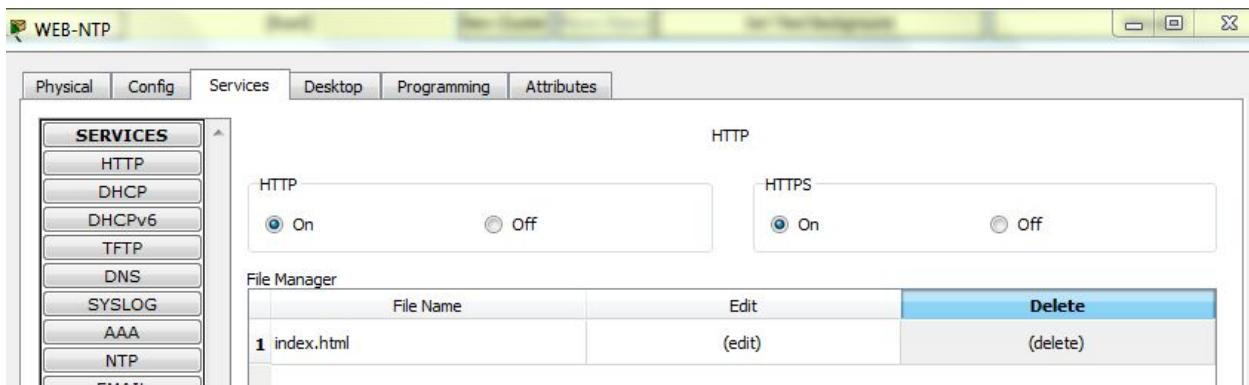
```
TIGRE(config)#logging host 10.10.45.2
TIGRE(config)#servi
TIGRE(config)#service ti
TIGRE(config)#service timestamps lo
TIGRE(config)#service timestamps log d
TIGRE(config)#service timestamps log datetime m
TIGRE(config)#service timestamps log datetime msec
TIGRE(config)#service timestamps log datetime msec
TIGRE(config)#service timestamps log datetime msec
TIGRE(config)#
TIGRE(config)#
TIGRE(config)#
TIGRE(config)#us
TIGRE(config)#username adm
TIGRE(config)#exit
TIGRE#
*may 25, 13:16:22.1616: SYS-5-CONFIG_I: Configured from console
by console
*may 25, 13:16:22.1616: %SYS-6-LOGGINGHOST_STARTSTOP: Logging to
host 10.10.45.2 port 514 started - CLI initiated
TIGRE#
```

Y ya lo tendremos configurado:

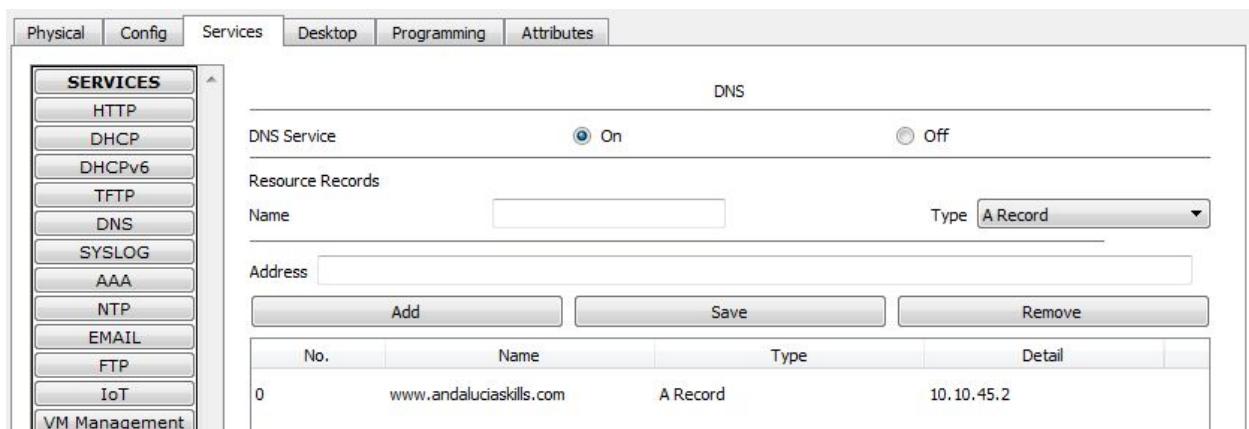
Time	HostName	Message
1 05.25.2018 01:16:22.953	10.10.45.1	%SYS-5-CONFIG_I: Configured from console by console
2 05.25.2018 01:16:22.953	10.10.45.1	: %SYS-6-LOGGINGHOST_STARTSTOP: Logging to host 10.10.45.2 port 514 s...

Lo siguiente que vamos a hacer ahora, es configurar el Servidor Web y DNS.

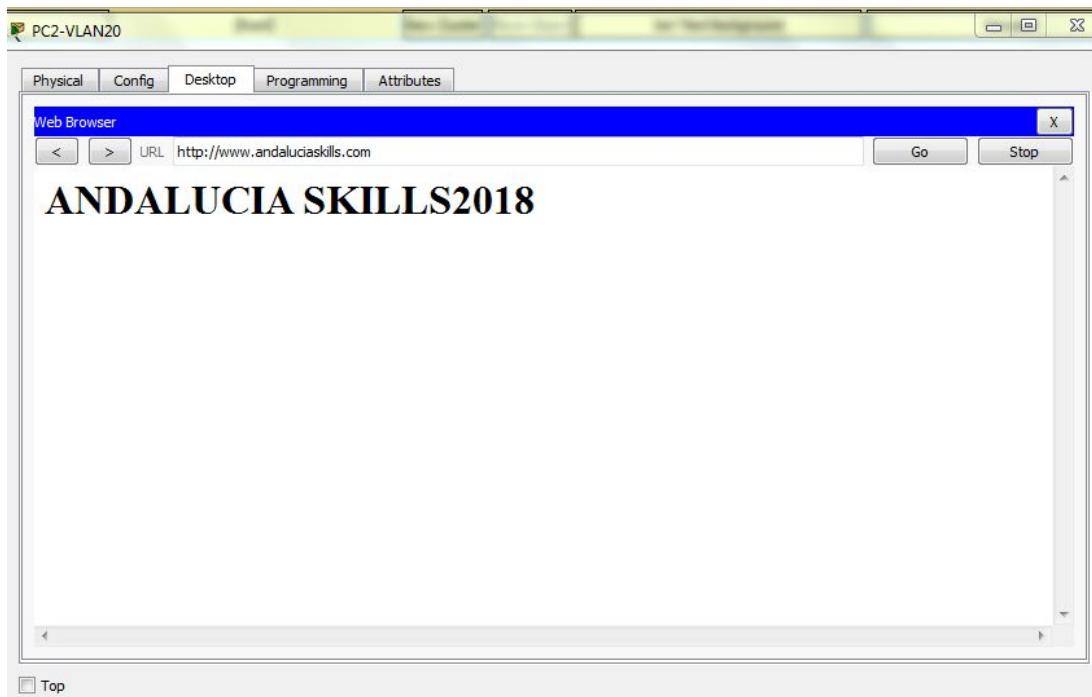
Para ello, nos iremos a los servidores y en primer lugar habilitaremos el web.



Ahora pasamos a configurar el DNS, que se encuentra en otro servidor:



Al realizar la prueba vemos como funciona correctamente



Ahora, pasaremos a habilitar ssh en los routers. Para ello, nos vamos a un router y debemos hacer lo siguiente.

```
TIGRE(config)#ip domain name tigre
TIGRE(config)#username tigre password cisco
TIGRE(config)#crypto key generate rsa
The name for the keys will be: TIGRE.tigre
Choose the size of the key modulus in the range of 360 to 2048
for your
    General Purpose Keys. Choosing a key modulus greater than 512
may take
    a few minutes.

How many bits in the modulus [512]: 1024
* Generating 1024 bit RSA keys, keys will be non-exportable...
[OK]

TIGRE(config)#ip ssh version 2
*may 30 11:37:22.86: %SSH-5-ENABLED: SSH 1.99 has been enabled
TIGRE(config)#

```

Una vez tengamos configurado SSH en los routers, sólo nos quedará impedir la conexión de los PC's de la red 10.10.10.0 al servidor web. Para ello, usaremos una ACL en el router que se encuentra unido al Servidor Web, que sería, el router TIGRE.

la acl sería la siguiente:

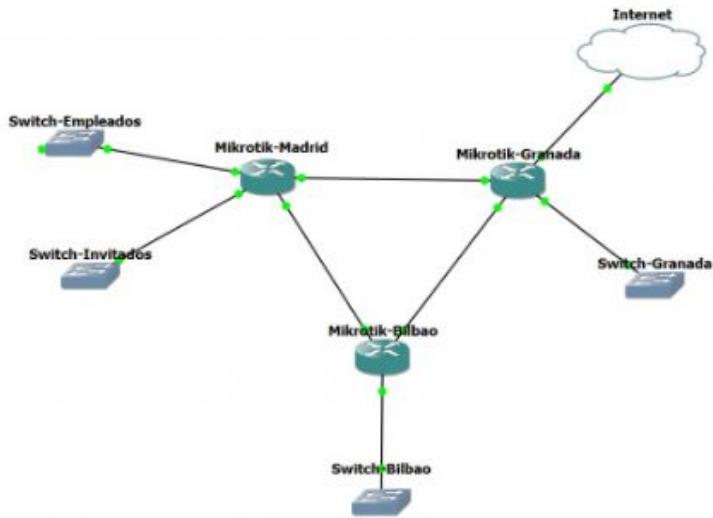
```
TIGRE(config)#access-list 10 deny 10.10.10.0 0.0.0.255
TIGRE(config)#do wr
Building configuration...
[OK]
TIGRE(config)#

```

Con esto, no permitimos el paso de todo el tráfico de la red 10.10.10.0/24 al Servidor Web. Con esto tendríamos terminada la parte de Cisco.

2. MikroTik test

Comenzamos la prueba de Mikrotik. En ella, tenemos el siguiente esquema de red:



Lo primero que debemos hacer es configurar las bocas de los routers y añadir la IP correspondiente.

En primer lugar, cambiaremos los nombres a los routers. Para ello usaremos el siguiente comando:

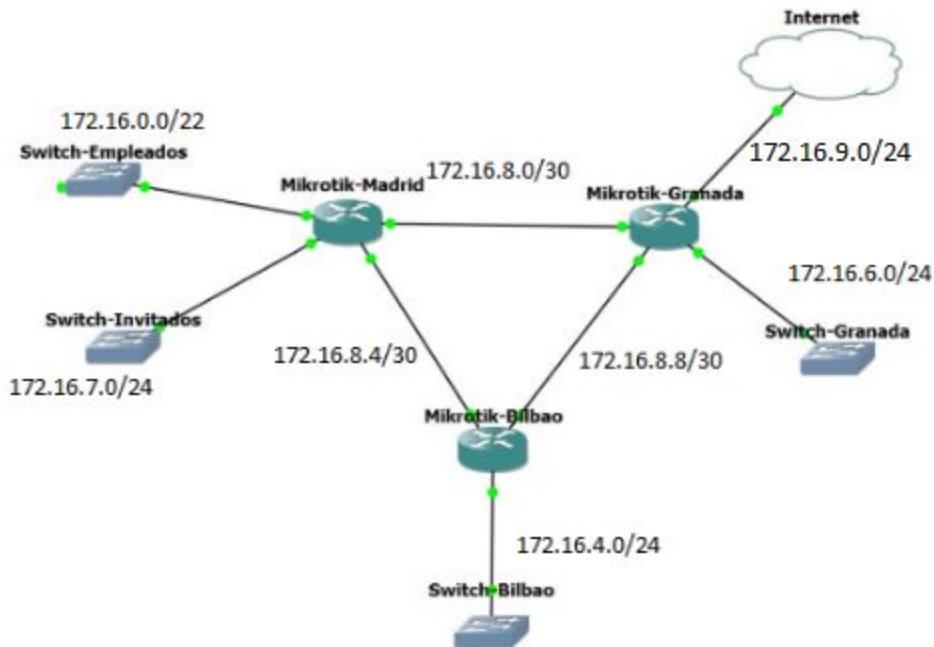
```
[admin@Mikrotik]>/system identity
[admin@Mikrotik]>/system identity set name = Mikrotik-Bilbao
[admin@Mikrotik]>/system identity set name = Mikrotik-Granada
[admin@Mikrotik]>/system identity set name = Mikrotik-Madrid
```

Ahora, el siguiente paso, vamos a hacer subnetting con la red proporcionada, que es la 172.16.0.0/16. Lo que vamos a hacer es determinar el número de redes que vamos a emplear y la máscara de red dependiendo del número de host en cada red.

Para ello vamos a usar las siguientes redes:

Red	Nombre de la Red	Máscara	Nº Máximo de Hosts
172.16.0.0/22	Madrid Empleados	255.255.252.0	1022
172.16.4.0/23	Bilbao	255.255.254.0	510
172.16.6.0/24	Granada	255.255.255.0	254
172.16.7.0/24	Madrid Invitados	255.255.255.0	254
172.16.8.0/30	Madrid-Granada	255.255.255.252	2
172.16.8.4/30	Madrid-Bilbao	255.255.255.252	2
172.16.8.8/30	Bilbao-Granada	255.255.255.252	2
172.16.9.0/24	Internet	255.255.255.0	254

El esquema de la red quedaría de la siguiente forma:



Con esto, tendríamos el planteamiento de la red preparada. Ahora, pasaremos a configurar los puertos de cada Router asignando las IP's correspondientes de la siguiente forma:

Router Mikrotik-Madrid:

```
[admin@Mikrotik-Madrid]> ip address add interface=ether1 address=172.16.0.1/22
[admin@Mikrotik-Madrid]> ip address add interface=ether2 address=172.16.8.1/30
[admin@Mikrotik-Madrid]> ip address add interface=ether3 address=172.16.7.1/24
[admin@Mikrotik-Madrid]> ip address add interface=ether4 address=172.16.8.5/30
```

Router Mikrotik-Bilbao:

```
[admin@Mikrotik-Bilbao]> ip address add interface=ether1 address=172.16.8.6/30
[admin@Mikrotik-Bilbao]> ip address add interface=ether2 address=172.16.8.9/30
[admin@Mikrotik-Bilbao]> ip address add interface=ether3 address=172.16.4.1/23
```

Router Mikrotik-Granada:

```
[admin@Mikrotik-Granada]> ip address add interface=ether1 address=172.16.8.2/30
[admin@Mikrotik-Granada]> ip address add interface=ether2 address=172.16.6.1/24
[admin@Mikrotik-Granada]> ip address add interface=ether3 address=172.16.8.10/30
[admin@Mikrotik-Granada]> ip address add interface=ether4 address=172.16.9.1/24
```

Daremos Ip a los puertos de cada router usando ipv6. Hemos hecho subnetting a la red proporcionada (2001:DB8:FE0::/48) y la hemos dividido en 8 subredes de modo que nos quedaría:

Red	Nombre de la Red	Nº Máximo de Hosts
2001:db8:fe0::/51	Madrid Empleados	8192
2001:db8:fe0:2000::/51	Bilbao	8192
2001:db8:fe0:4000::/51	Granada	8192
2001:db8:fe0:6000::/51	Madrid Invitados	8192
2001:db8:fe0:8000::/51	Madrid-Granada	8192

2001:db8:fe0:a000::/51	Madrid-Bilbao	8192
2001:db8:fe0:c000::/51	Bilbao-Granada	8192
2001:db8:fe0:e000::/51	Internet	8192

Router Madrid

```
[admin@Mikrotik-Madrid]> ipv6 add address interface=ether1
address=2001:db8:fe0::1/51
```

```
[admin@Mikrotik-Madrid]> ipv6 add address interface=ether2
address=2001:db8:fe0:8000:1/51
```

```
[admin@Mikrotik-Madrid]> ipv6 add address interface=ether3
address=2001:db8:fe0:6000:1/51
```

```
[admin@Mikrotik-Madrid]> ipv6 add address interface=ether4
address=2001:db8:fe0:a000:1/51
```

Router Bilbao

```
[admin@Mikrotik-Bilbao]> ipv6 add address interface=ether1
address=2001:db8:fe0:a000:2/51
```

```
[admin@Mikrotik-Bilbao]> ipv6 add address interface=ether2
address=2001:db8:fe0:c000:2/51
```

```
[admin@Mikrotik-Bilbao]> ipv6 add address interface=ether3
address=2001:db8:fe0:2000:1/51
```

Router Granada

```
[admin@Mikrotik-Granada]> ipv6 add address interface=ether1
address=2001:db8:fe0:8000:2/51
```

```
[admin@Mikrotik-Granada]> ipv6 add address interface=ether2
address=2001:db8:fe0:4000:1/51
```

```
[admin@Mikrotik-Granada]> ipv6 add address interface=ether3
address=2001:db8:fe0:c000:1/51
```

```
[admin@Mikrotik-Granada]> ipv6 add address interface=ether4
address=2001:db8:fe0:e000:1/51
```

Una vez configurados los puertos de cada router, vamos a pasar a habilitar el OSPF.

Para identificar los routers, vamos a crear una interfaz loopback en cada uno de los router. Lo haremos en la Red 172.16.255.0/24.

Esta interfaz loopback también sirve para evitar que la identificación del router se asocie a una interfaz física y así, si estuviera asociada a una interfaz física y sufriera una caída no provocaría problemas en la identificación entre routers.

De este modo, como la interfaz loopback siempre está activa no se produce tal problema.

Router Madrid:

```
[admin@Mikrotik-Madrid]> interface bridge add name=loopback
[admin@Mikrotik-Madrid]> ip address add interface=loopback
address=172.16.255.1/32
```

Router Granada:

```
[admin@Mikrotik-Granada]> interface bridge add name=loopback
[admin@Mikrotik-Granada]> ip address add interface=loopback
address=172.16.255.2/32
```

Router Bilbao

```
[admin@Mikrotik-Bilbao]> interface bridge add name=loopback
[admin@Mikrotik-Bilbao]> ip address add interface=loopback address=172.16.255.3/32
```

Pasamos a configurar la instancia de OSPF asociando también la identificación del router que generamos creando la interfaz loopback con su direccionamiento.

```
[admin@Mikrotik-Madrid]> routing ospf instance set default router-id=172.16.255.1
redistribute-connected=as-type-1
```

```
[admin@Mikrotik-Granada]> routing ospf instance set default router-id=172.16.255.2
redistribute-connected=as-type-1
```

```
[admin@Mikrotik-Bilbao]> routing ospf instance set default router-id=172.16.255.3
redistribute-connected=as-type-1
```

Ahora, en cada router lo que nos queda es hacer que publiquen cada red a la que ellos pertenecen.

Router Madrid:

```
[admin@Mikrotik-Madrid]> routing ospf network add network=172.16.0.0/22  
area=backbone  
[admin@Mikrotik-Madrid]> routing ospf network add network=172.16.8.0/30  
area=backbone  
[admin@Mikrotik-Madrid]> routing ospf network add network=172.16.7.0/24  
area=backbone  
[admin@Mikrotik-Madrid]> routing ospf network add network=172.16.8.4/30  
area=backbone
```

Router Granada:

```
[admin@Mikrotik-Granada]> routing ospf network add network=172.16.6.0/24  
area=backbone  
[admin@Mikrotik-Granada]> routing ospf network add network=172.16.8.0/30  
area=backbone  
[admin@Mikrotik-Granada]> routing ospf network add network=172.16.8.8/30  
area=backbone  
[admin@Mikrotik-Granada]> routing ospf network add network=172.16.9.0/24  
area=backbone
```

Router Bilbao

```
[admin@Mikrotik-Bilbao]> routing ospf network add network=172.16.4.0/23  
area=backbone  
[admin@Mikrotik-Bilbao]> routing ospf network add network=172.16.8.4/30  
area=backbone  
[admin@Mikrotik-Bilbao]> routing ospf network add network=172.16.8.8/30  
area=backbone
```

Ahora, ya tendremos listo el enrutamiento y podremos enviar paquetes de extremo a extremo sin problemas.

Ahora, vamos a habilitar la ruta a internet por defecto. Para ello, hemos puesto la ruta de internet en el Router **Mikrotik-Granada**.

```
[admin@Mikrotik-Granada]> ip route add dst-address=0.0.0.0/0 gateway=172.16.9.1
[admin@Mikrotik-Granada]> routing ospf instance set default
distribute-default-if-installed-as-type-1
```

Ahora, vamos a configurar el router de Granada para que los usuarios puedan acceder a internet usando una dirección pública. Para ello, usaremos el protocolo NAT para enmascarar las direcciones IP privadas y convertirlas en públicas. (Nosotros, estamos simulando la red 172.16.9.0/24 como red pública, la IP pública te la proporciona el proveedor de internet.)

```
[admin@Mikrotik-Granada]> ip firewall nat add chain=srcnat src-address=172.16.6.0/24
dst-address=0.0.0.0/0 out-interface=ether4 action=masquerade disabled=no
```

El siguiente paso a realizar será habilitar el DHCP en el Router **Mikrotik-Granada** y habilitar DHCP Relay en los demás routers para que dejen pasar los paquetes DHCP. Para ello, nos dirigimos a el Mikrotik-Granada y comenzamos por habilitar los pool de DHCP para las distintas redes.

```
[admin@Mikrotik-Granada]> ip pool add name=GRA ranges=172.16.6.2-172.16.6.254
[admin@Mikrotik-Granada]> ip pool add name=BILB ranges=172.16.4.2-172.16.5.254
[admin@Mikrotik-Granada]> ip pool add name=MADE ranges=172.16.3.2-172.16.3.254
[admin@Mikrotik-Granada]> ip pool add name=MADI ranges=172.16.7.2-172.16.7.254
```

Una vez hayamos añadido todos los pools, lo siguiente que vamos a hacer es crear los servidores DHCP para cada una de las bocas del router correspondientes para cada Red.

```
[admin@Mikrotik-Granada]> ip dhcp-server add name=GRA interface=ether2
address-pool=GRA lease-time=259200 bootp-lease-time=forever authoritative=yes
disabled=no
[admin@Mikrotik-Granada]> ip dhcp-server add name=BILB interface=ether3
address-pool=BILB lease-time=259200 bootp-lease-time=forever authoritative=yes
disabled=no
[admin@Mikrotik-Granada]> ip dhcp-server add name=MADE interface=ether1
address-pool=MADE lease-time=259200 bootp-lease-time=forever authoritative=yes
disabled=no
[admin@Mikrotik-Granada]> ip dhcp-server add name=MADI interface=ether1
address-pool=MADI lease-time=259200 bootp-lease-time=forever authoritative=yes
disabled=no
```

Ahora, ya tenemos el Servidor DHCP correctamente funcional. Ahora, pasaremos a configurar los DHCP Relay en cada uno de los routers para que puedan pasar los paquetes de DHCP a través de ellos.

Para ello, lo que hay que hacer es añadir los siguientes comandos en el puerto que van a ir destinados los paquetes DHCP.

Router Bilbao

```
[admin@Mikrotik-Bilbao]> ip dhcp-relay add name=Relay1 interface=ether3
dhcp-server=172.16.8.10 local-address=172.16.4.1 disabled=no
```

Router Madrid

```
[admin@Mikrotik-Madrid]> ip dhcp-relay add name=Relay1 interface=ether1
dhcp-server=172.16.8.2 local-address=172.16.0.1 disabled=no
[admin@Mikrotik-Madrid]> ip dhcp-relay add name=Relay1 interface=ether3
dhcp-server=172.16.8.2 local-address=172.16.7.1 disabled=no
```

Con esto, habremos terminado de configurar el DHCP y ya los equipos que se conecten en la red usarán una ip asignada por el Router **Mikrotik-Granada**.

Pasamos a configurar el HotSpot. Para ello, hay una forma fácil de hacerlo que es usando el propio setup de Mikrotik. con usar el comando siguiente nos irá pidiendo los parámetros que nosotros queramos usar para habilitar el Hotspot:

```
[admin@Mikrotik-Madrid]> ip hotspot setup
```

Y completamos con los siguientes datos:

- **Hotspot Interface:** ether3
- **Local Address of network:** 172.16.7.0/24
- **Masquerade:** yes
- **Address Pool of network:** 172.16.7.2-172.16.7.255
- **Select certificate:** none
- **IP address of SMTP server:** 0.0.0.0
- **DNS Servers:** 8.8.8.8,8.8.4.4
- **DNS name:** <vacío>
- **Name of local hotspot user:** admin
- **Password:** <contraseña>

Para usar SSL en nuestro Hotspot, lo primero que debemos hacer es generar la clave privada, lo haremos desde un pc linux:

```
root@ubuntu:/home/usuario# openssl genrsa -des3 -out hotspot.key 1024
```

Ahora, generamos el csr:

```
root@ubuntu:/home/usuario# openssl req -new -key hotspot.key -out hotspot.csr
```

Ahora, Generamos una solicitud de firma certificada (CSR) basada en un certificado existente:

```
root@ubuntu:/home/usuario# openssl x509 -req -days 10000 -in hotspot.csr -signkey hotspot.key -out hotspot.crt
```

Lo siguiente será subir los ficheros hotspot.key y hotspot.crt al router. Cuando estén subidos, con el siguiente comando los instalamos en nuestro router:

```
[admin@mikrotik-Madrid]> certificate import file-name=hotspot.crt  
[admin@mikrotik-Madrid]> certificate import file-name=hotspot.crt
```

Ahora, una vez importado correctamente el certificado y la clave privada en el enrutador, primero hay que habilitar el servicio ssl y agregar el nombre del certificado en el servicio / ip:

```
[admin@mikrotik-Madrid]> ip service set www-ssl certificate=cert1
```

Ahora, habilitamos HTTPS en nuestro Hotspot:

```
[admin@mikrotik-Madrid]> ip hotspot profile set hsprof1 login-by=https  
ssl-certificate=cert1
```

Para añadir usuarios en nuestro Hotspot usaremos el siguiente comando:

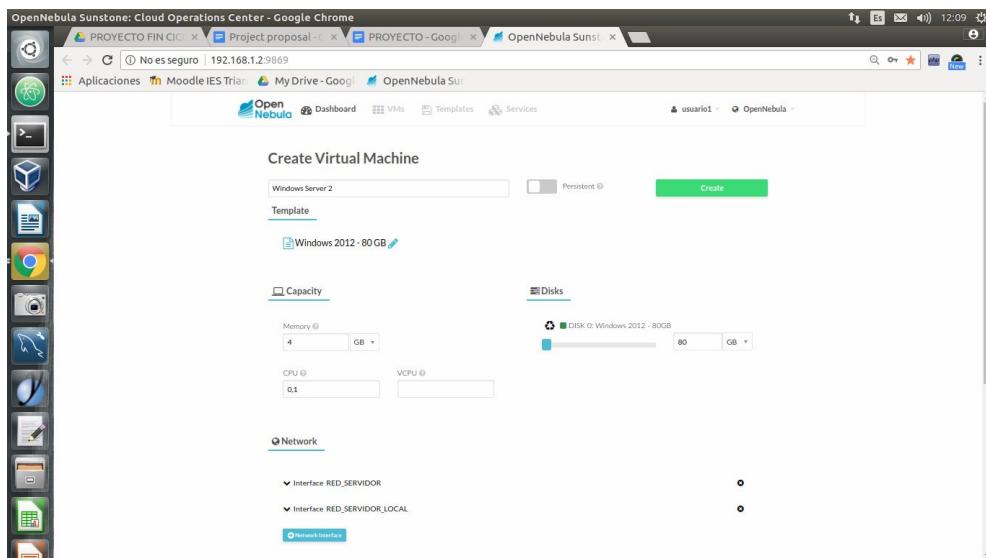
```
[admin@mikrotik-Madrid]> ip hotspot user add name=pedro password=123456789
```

3. Servicios

1. Creación Máquinas virtuales

Para este proyecto, vamos a usar un Cluster en el que desplegamos las máquinas virtuales usando Open Nebula.

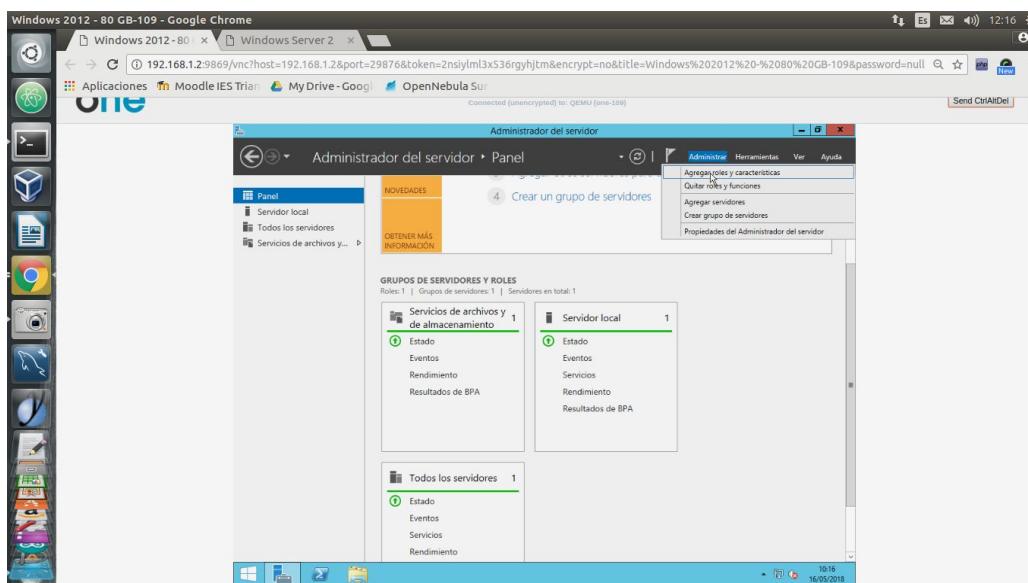
Aquí tenemos la pantalla de como vamos a desplegar los Windows Server:

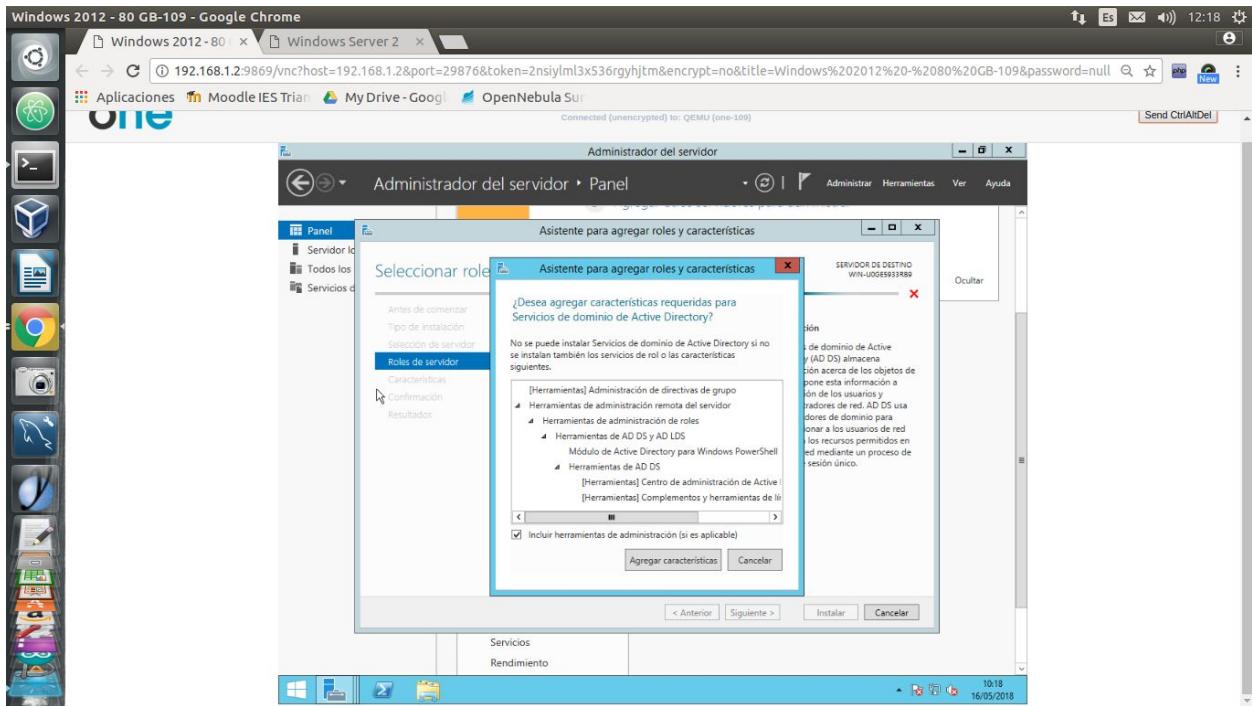


Para entrar en los Windows Server por defecto la contraseña de Administrador es **usuUSU123**.

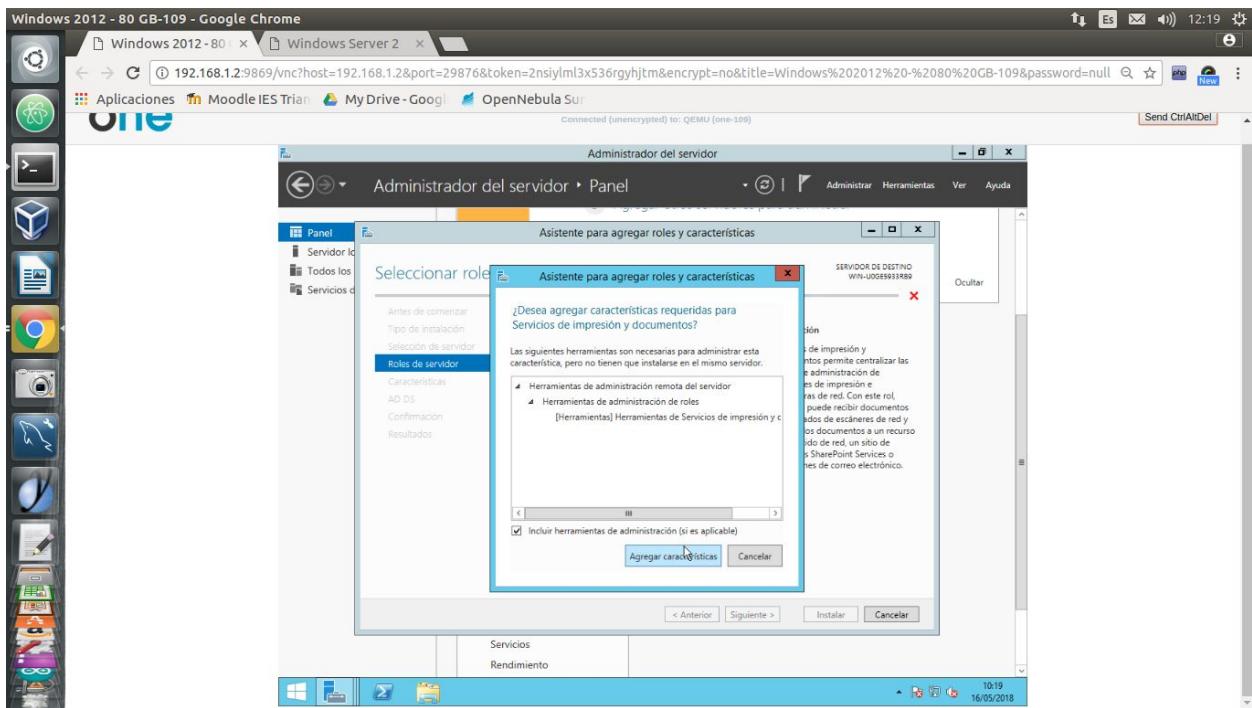
2. Active Directory y controlador Secundario.

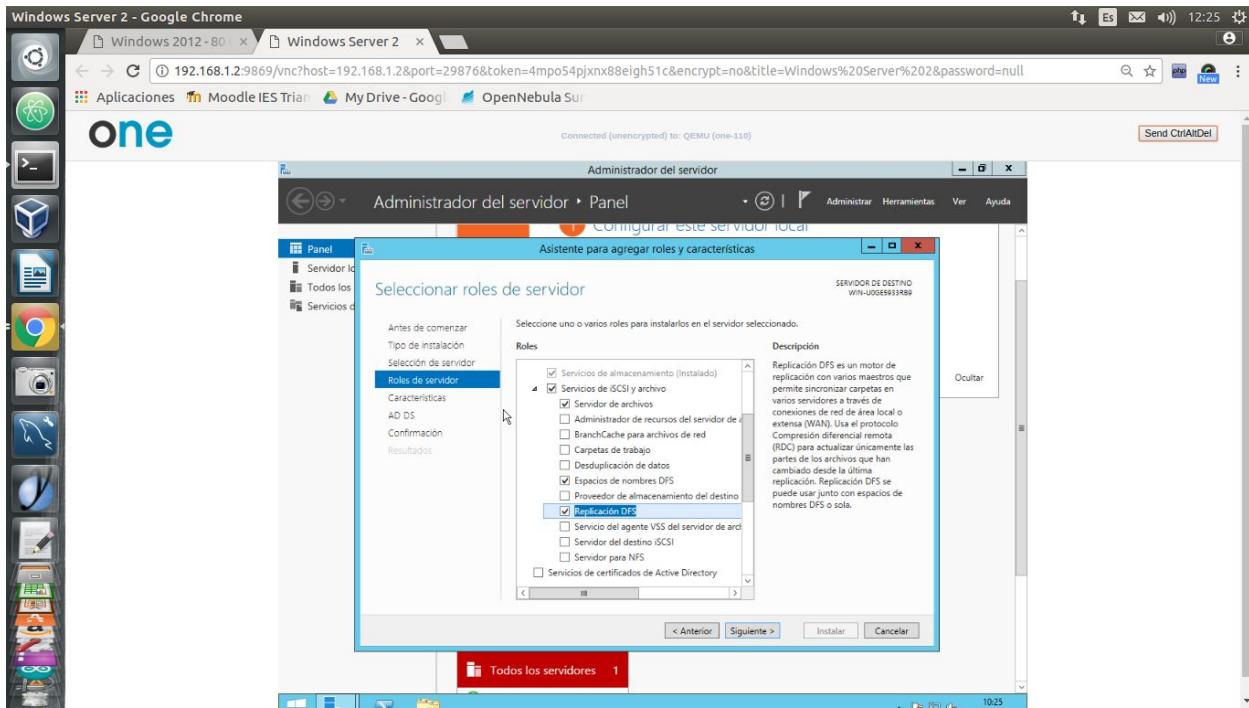
Ahora, pasaremos a crear el Active Directory. Lo primero que debemos hacer será añadir los roles y características necesarias.



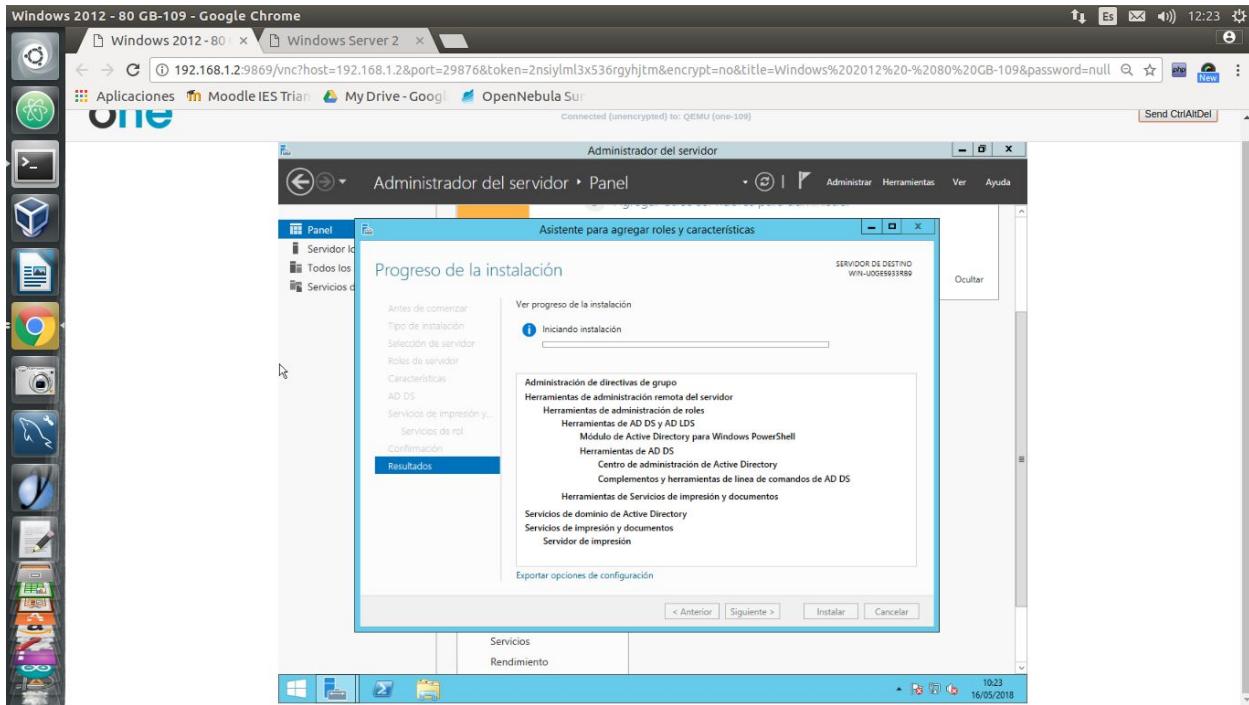


También instalaremos el servidor de impresión y el DFS:

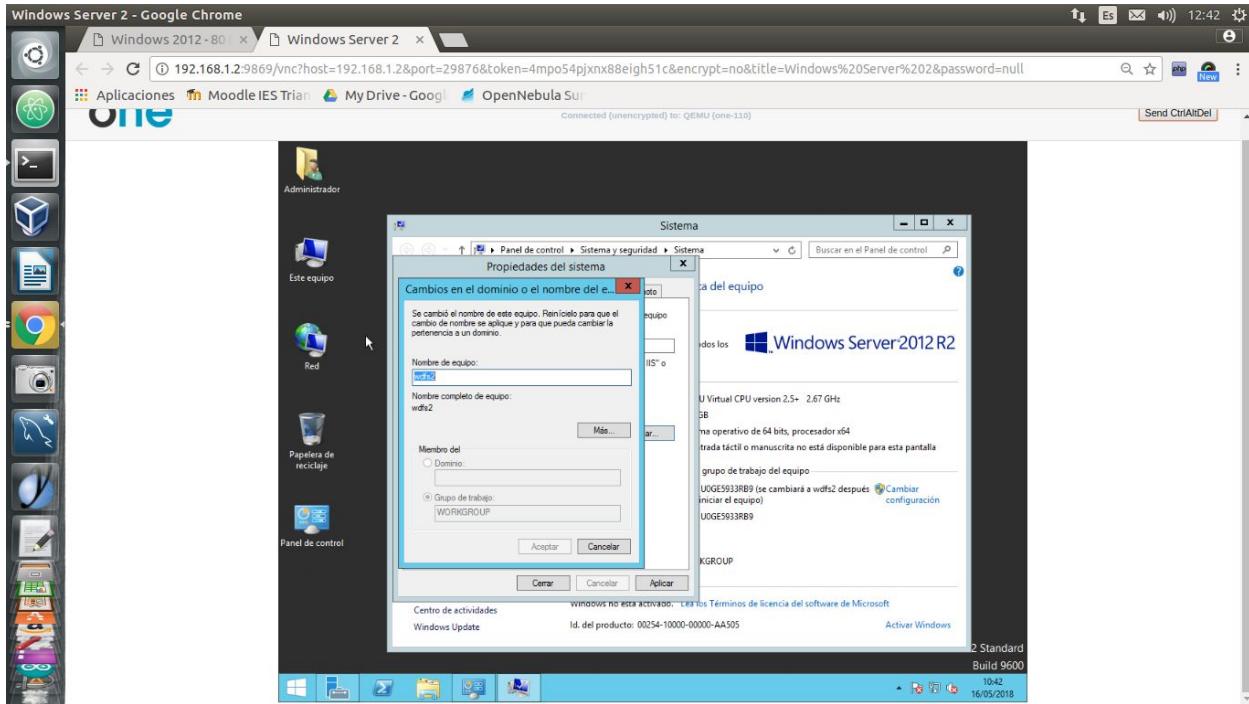




Ahora podemos proceder a la instalación de estos roles del Servidor, en el otro servidor deberemos hacer lo mismo pero sin Servidor de Impresión.

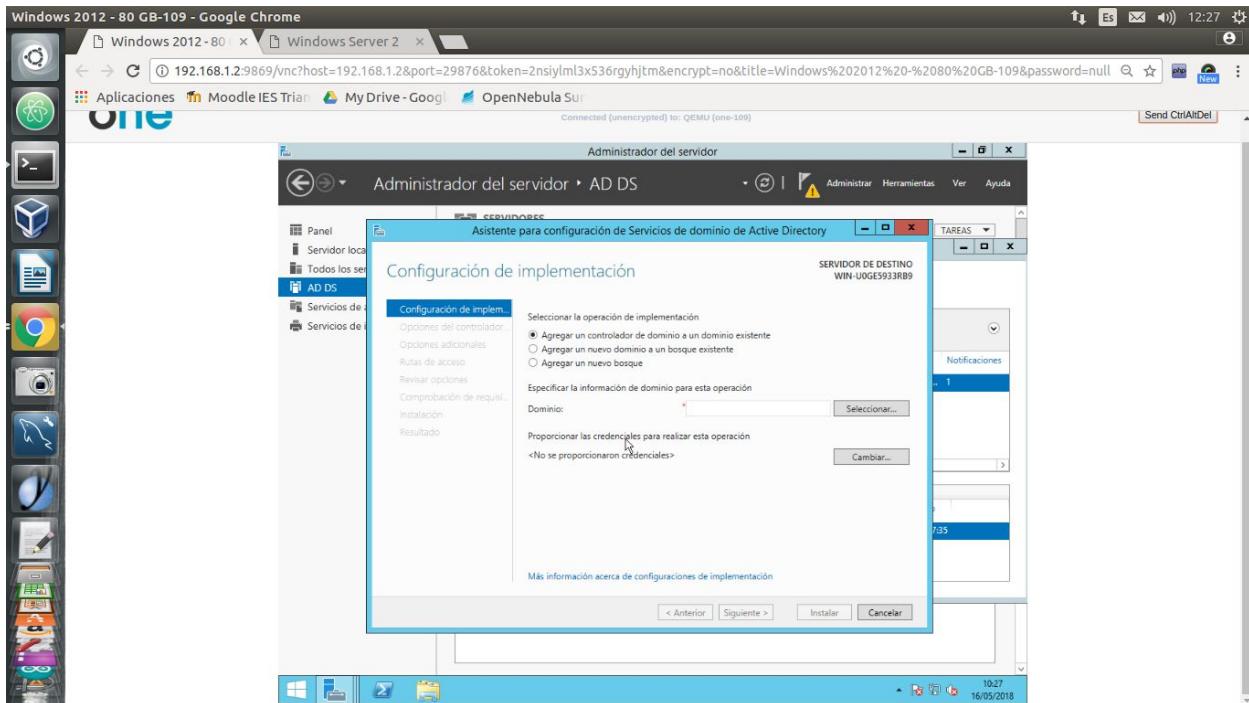


Lo siguiente que vamos a hacer antes de promover el primer Servidor como controlador de Dominio será cambiarle el nombre a las máquinas. Las llamaremos wdfs1 y wdfs2 y también asignaremos ip's estáticas.

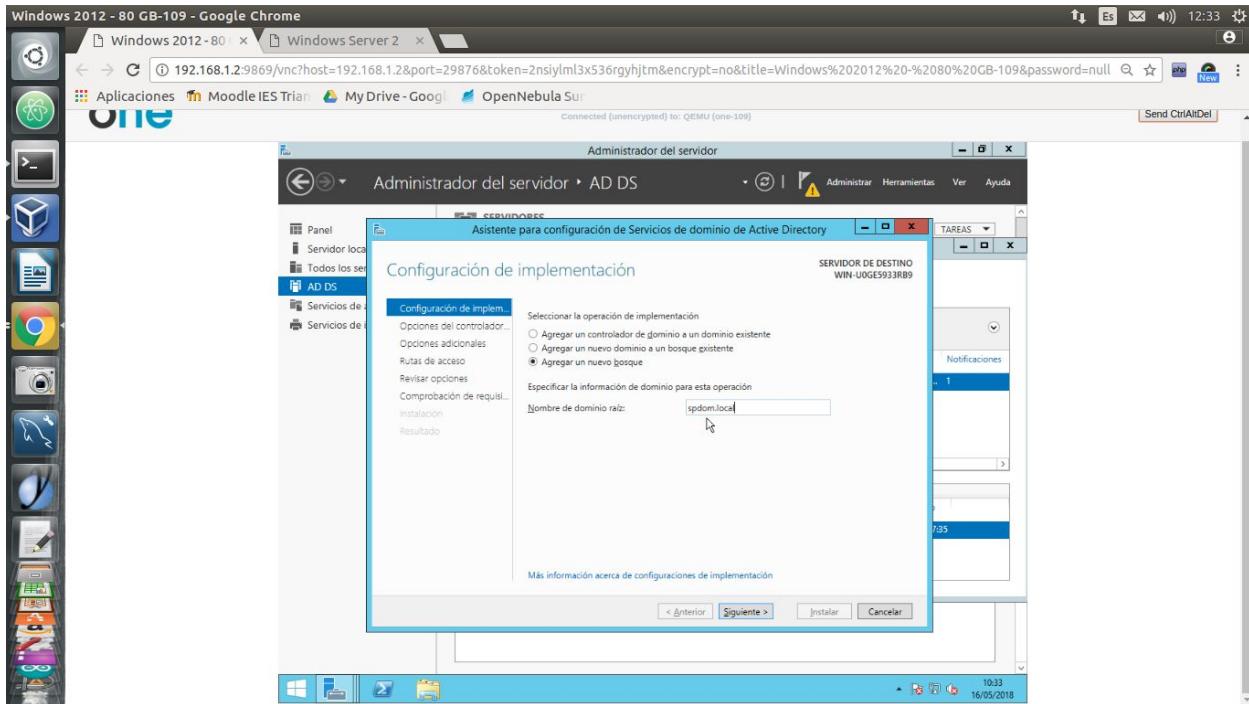


Ahora, lo siguiente que vamos a hacer será hacer el aprovisionamiento para usarla como controladora de dominio.

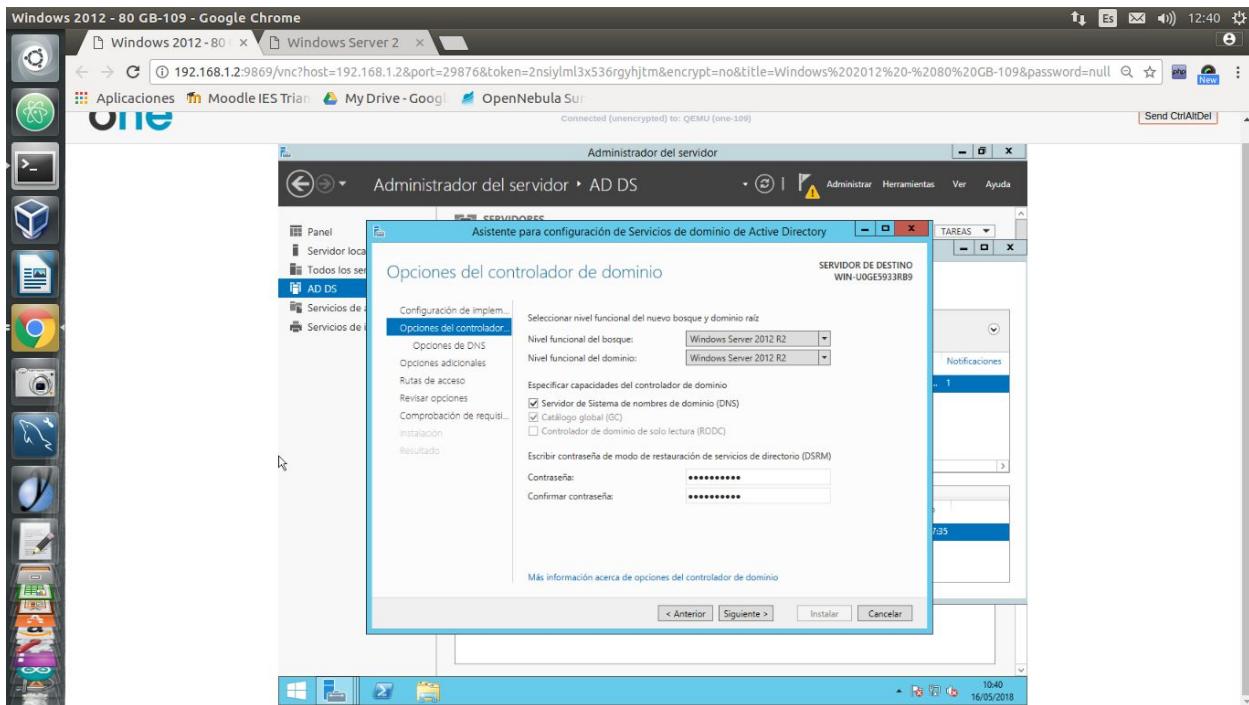
Para ello, nos dirigiremos a Administración del Servidor y lo realizaremos desde allí.



El nombre de nuestro dominio será **spdom.local**

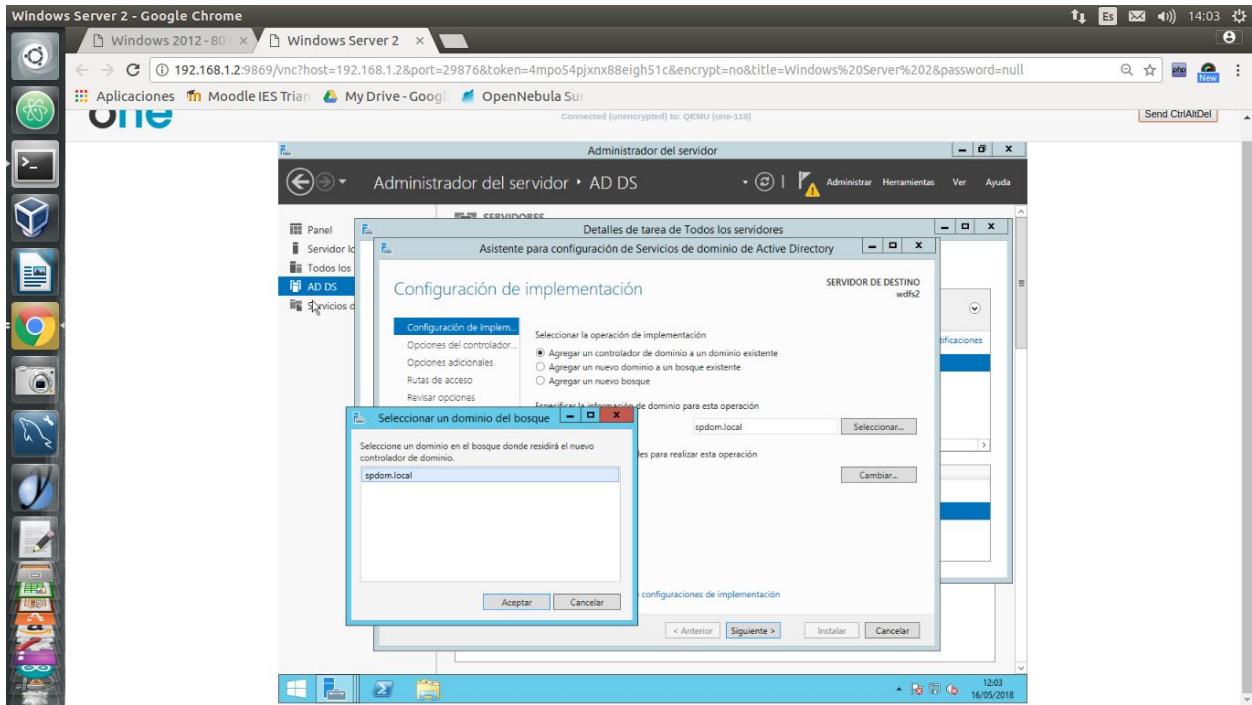


La contraseña de modo de restauración será **usuUSU123@**

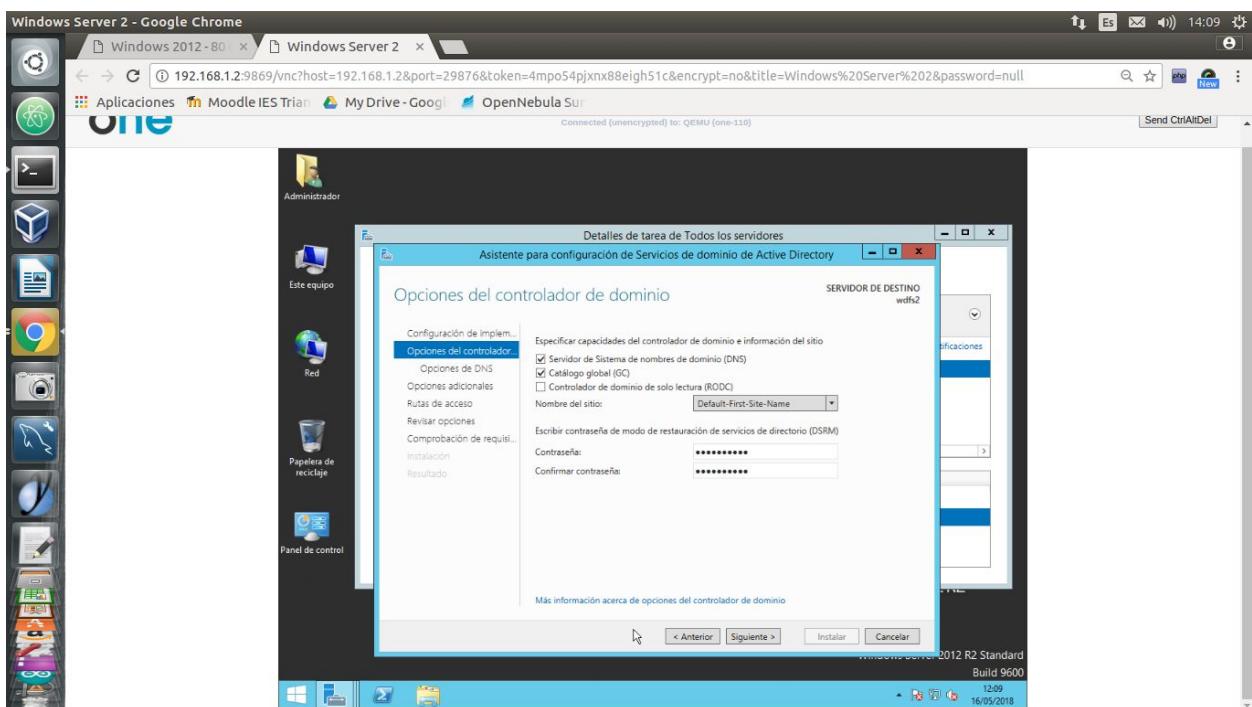


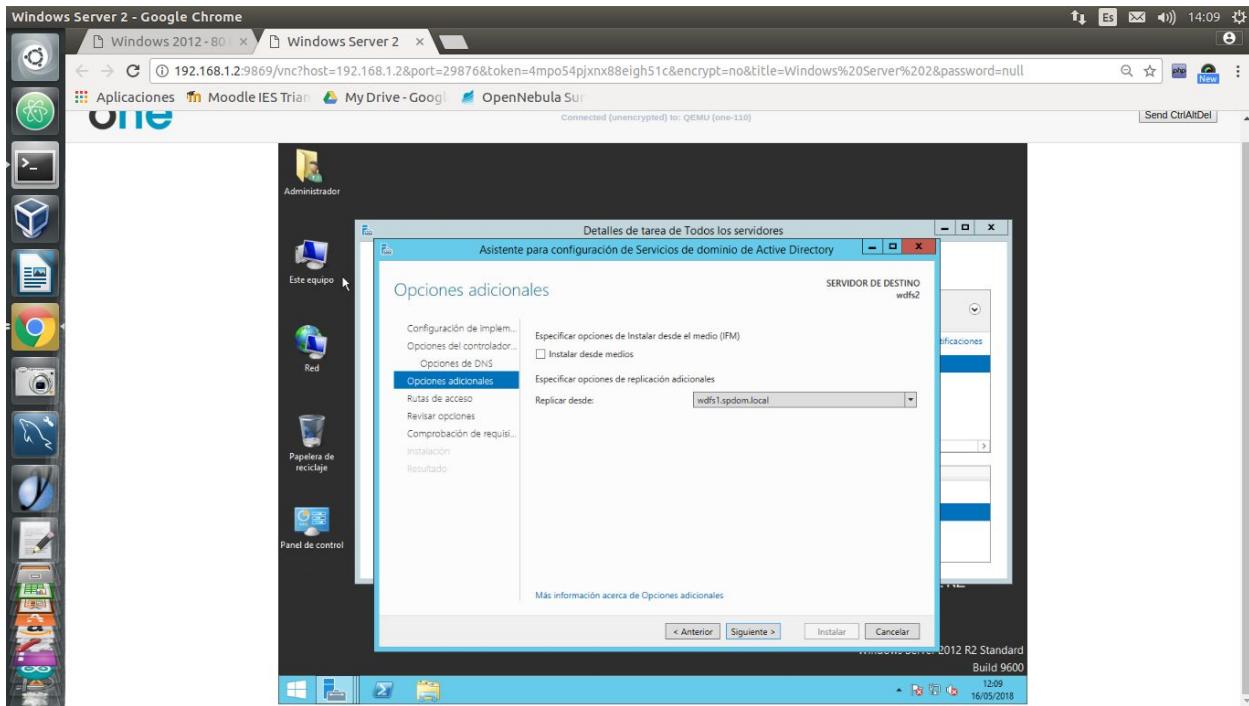
Confirmamos y comenzará a instalar. Una vez terminado el proceso de instalación el servidor deberá reiniciarse y iniciaremos sesión y ya lo tendremos como controlador de dominio. Lo siguiente que vamos a hacer será promover el segundo servidor como controlador de dominio secundario.

Lo que debemos hacer será promoverlo y deberemos especificar el dominio al que pertenece e iniciar sesión en el dominio con la cuenta de Administrador.

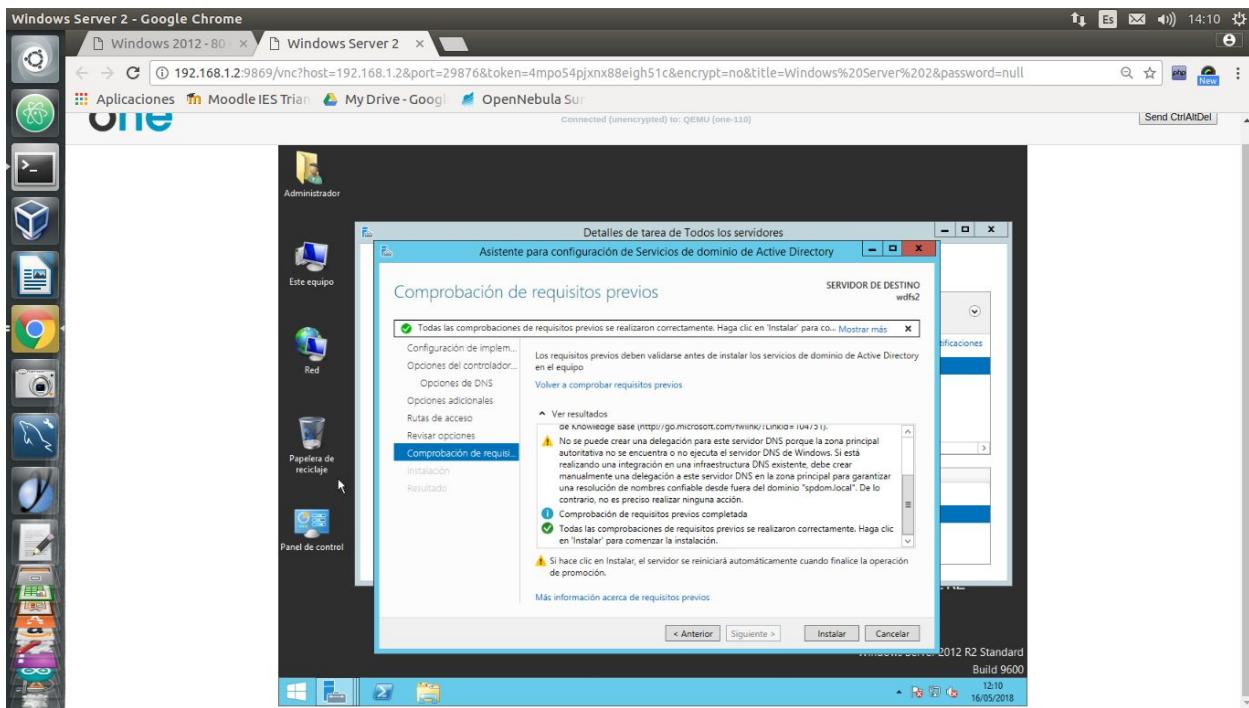


Continuamos con la instalación:





Clicamos en instalar y comenzará la instalación:

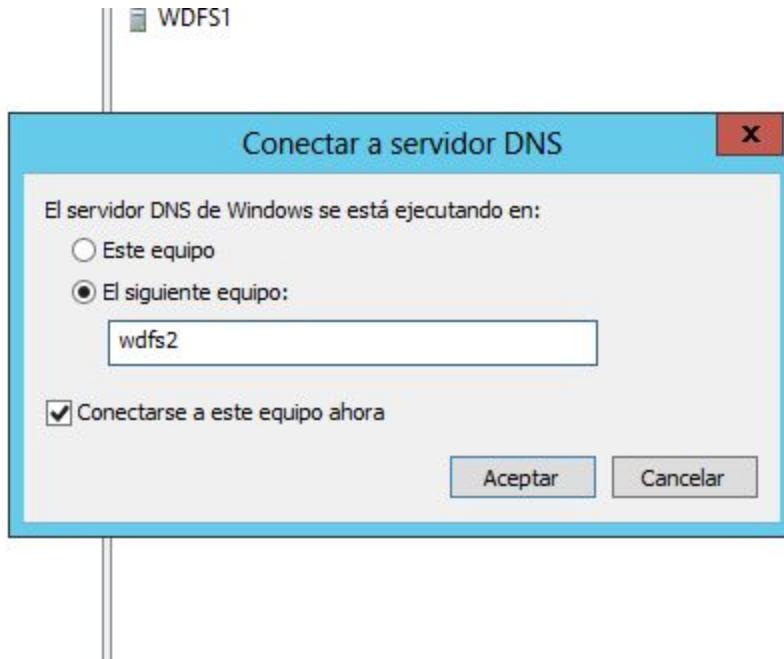


Una vez finalizado ya tendremos nuestro controlador de dominio secundario.

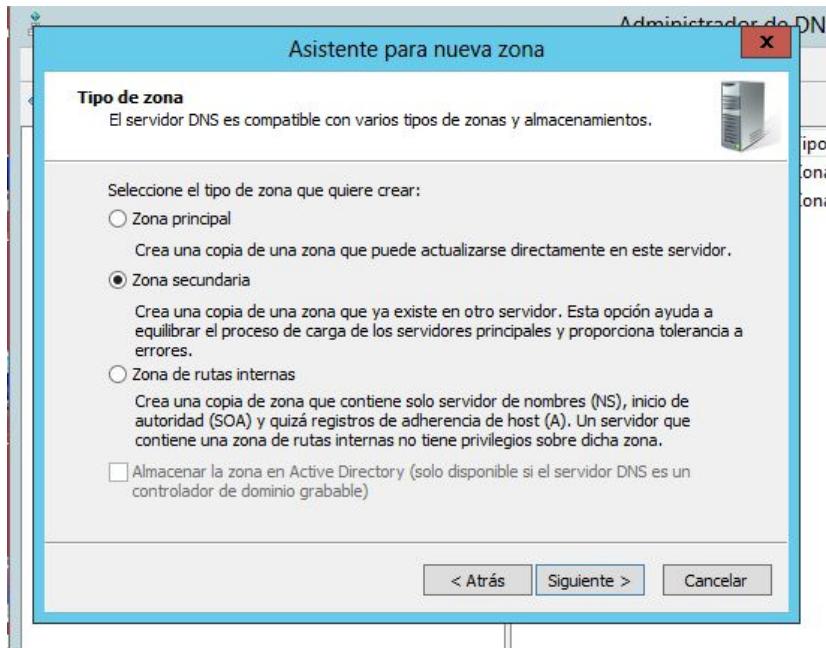
3. DNS

El siguiente paso será configurar los servidores DNS para tener un servidor DNS secundario, en este caso wdfs2. Para ello, nos dirigimos a wdfs1 y en el Administrador de DNS deberemos dar click derecho en DNS y damos a conectar con el servidor DNS.

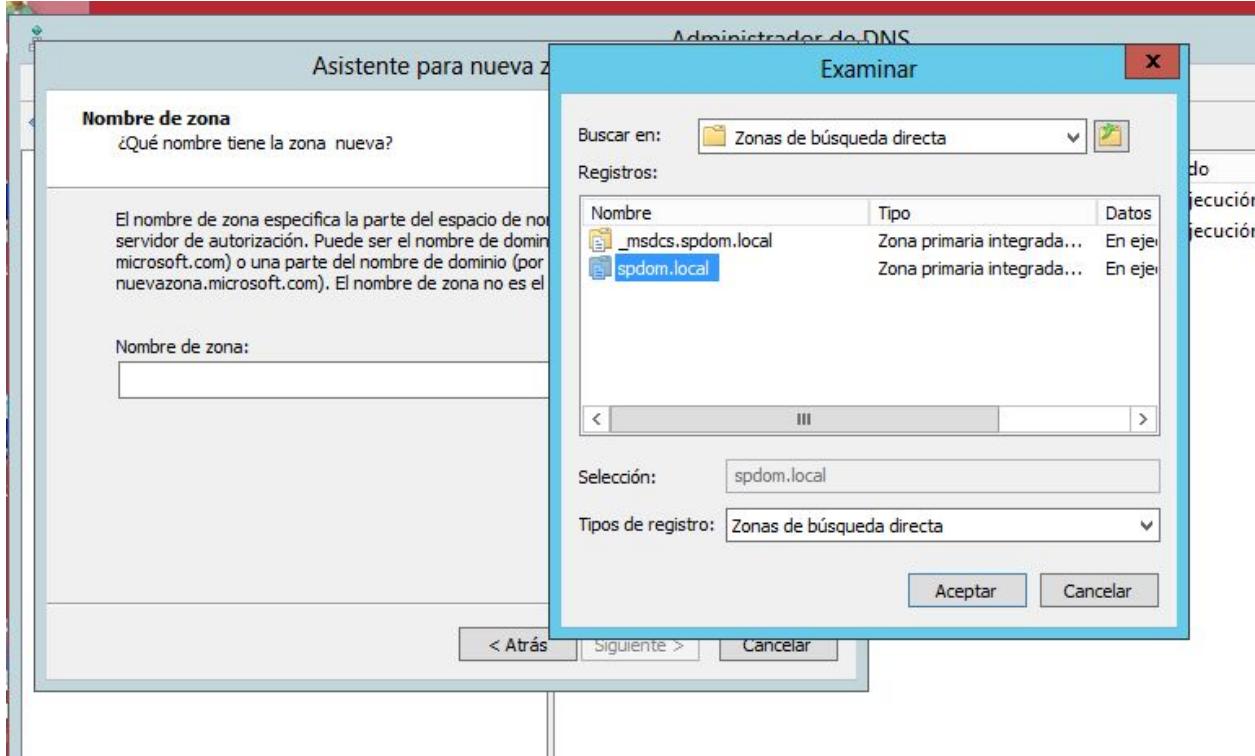
Ahora, ponemos el nombre de wdfs2 para que se conecte a él.



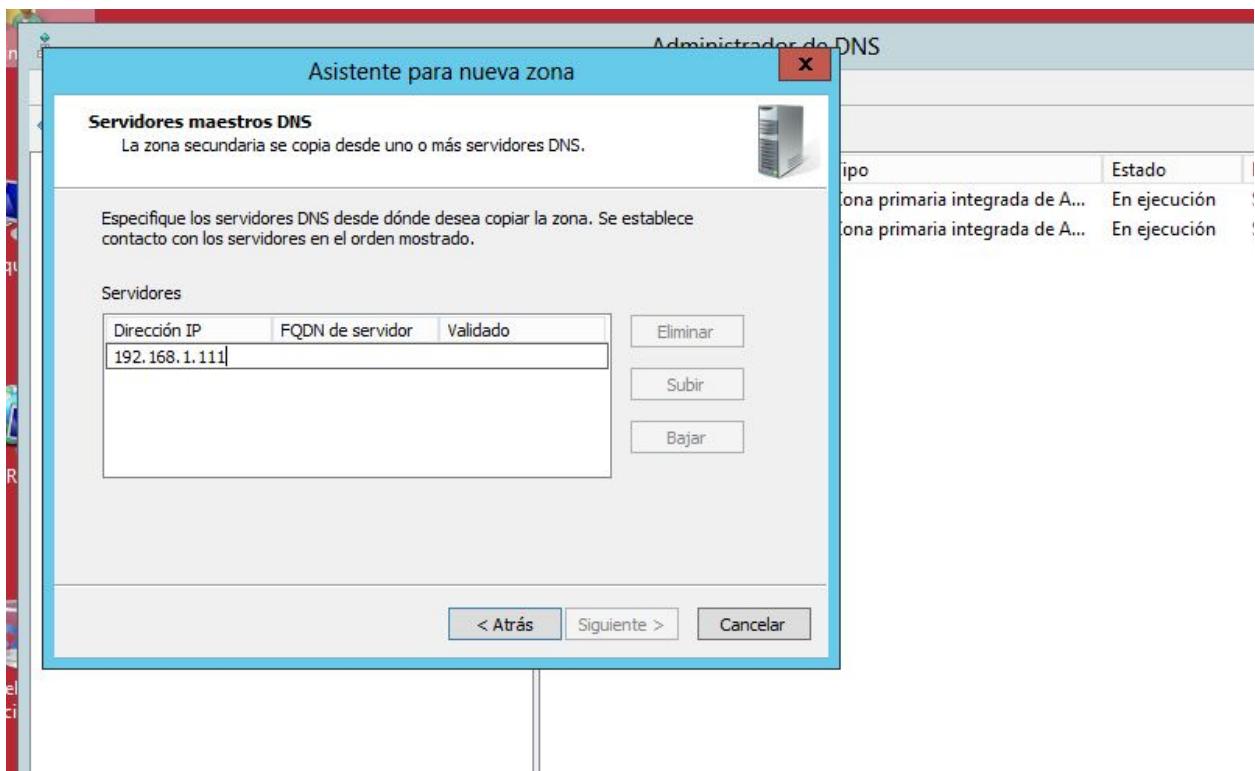
Hacemos click en aceptar. Lo siguiente que vamos a hacer es dar click derecho en zonas directa en wdfs2 y click en zona nueva y seleccionamos zona secundaria.

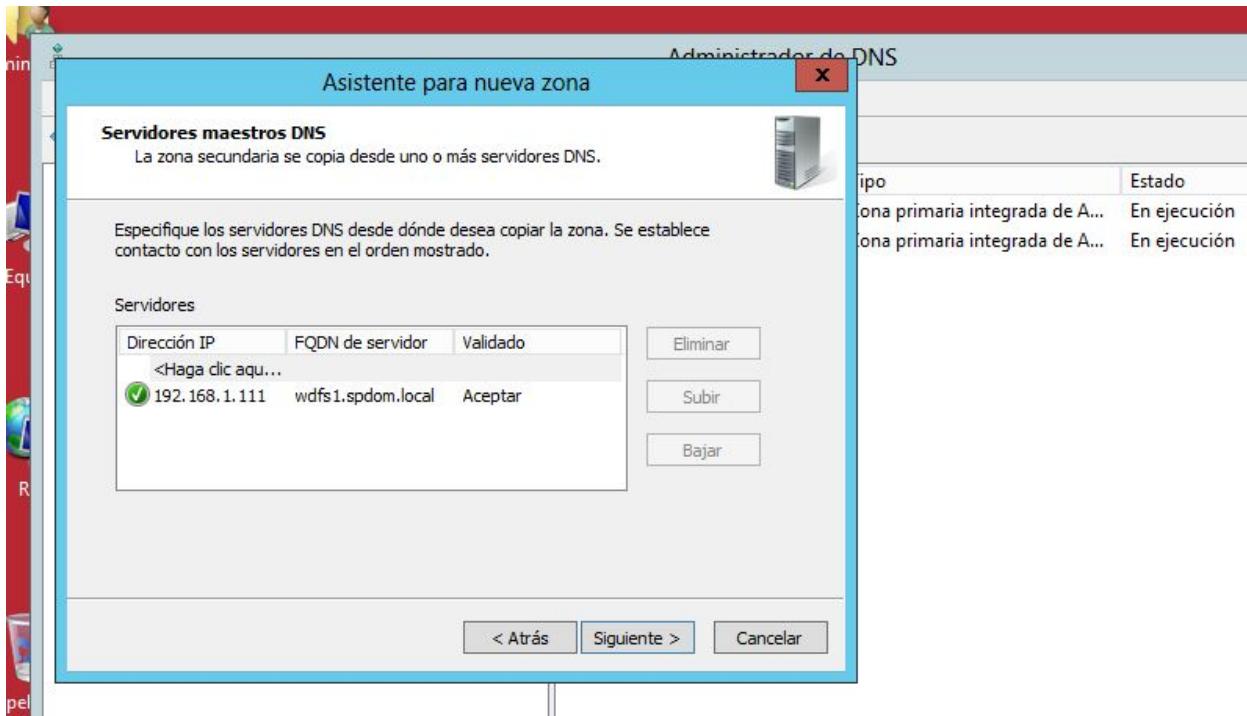


Ahora, elegimos nuestra zona de domio spdom.local:

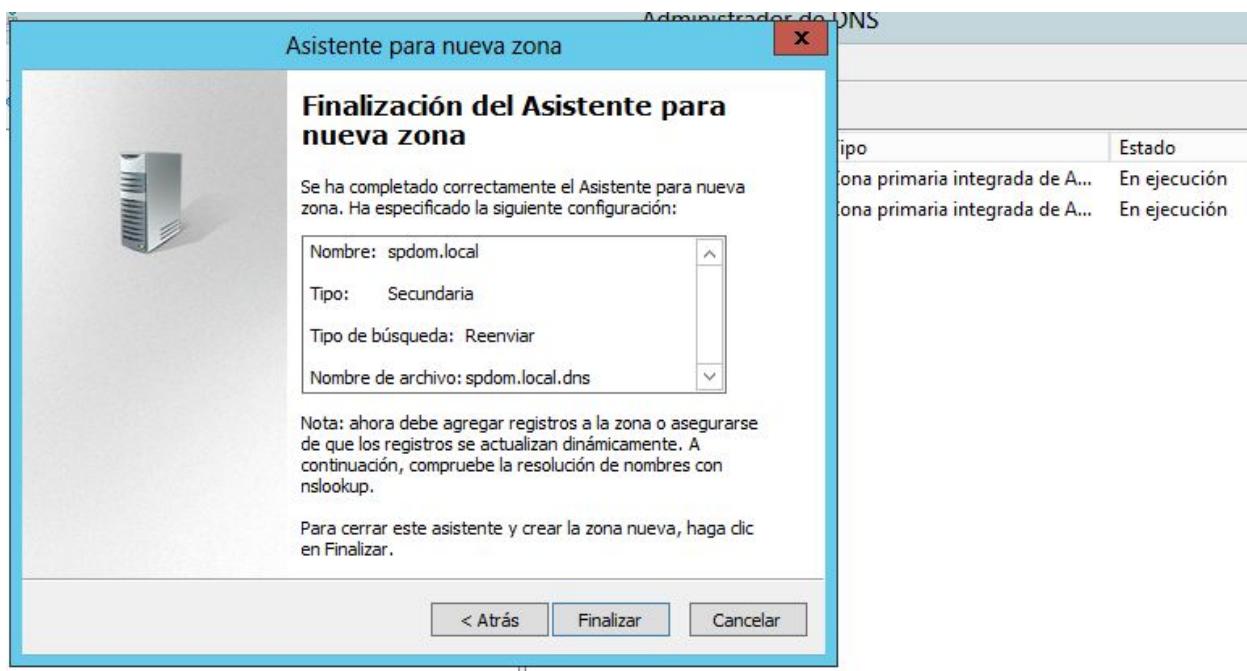


Y añadimos la ip de wdfs1:





Y con esto ya lo tenemos configurado:

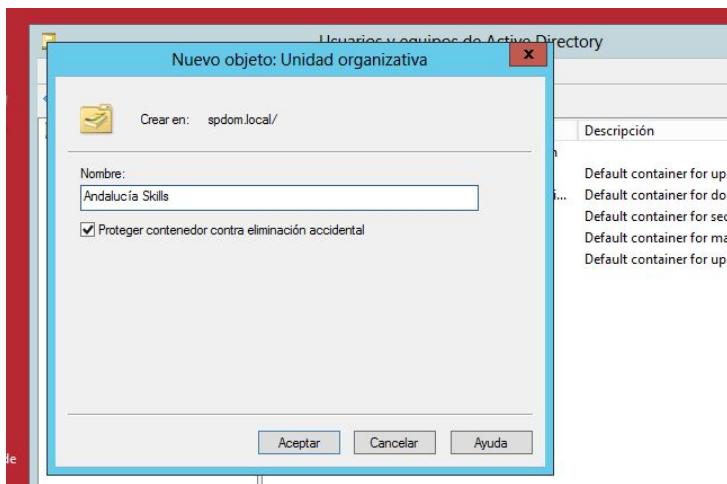


4. Usuarios del Active Directory.

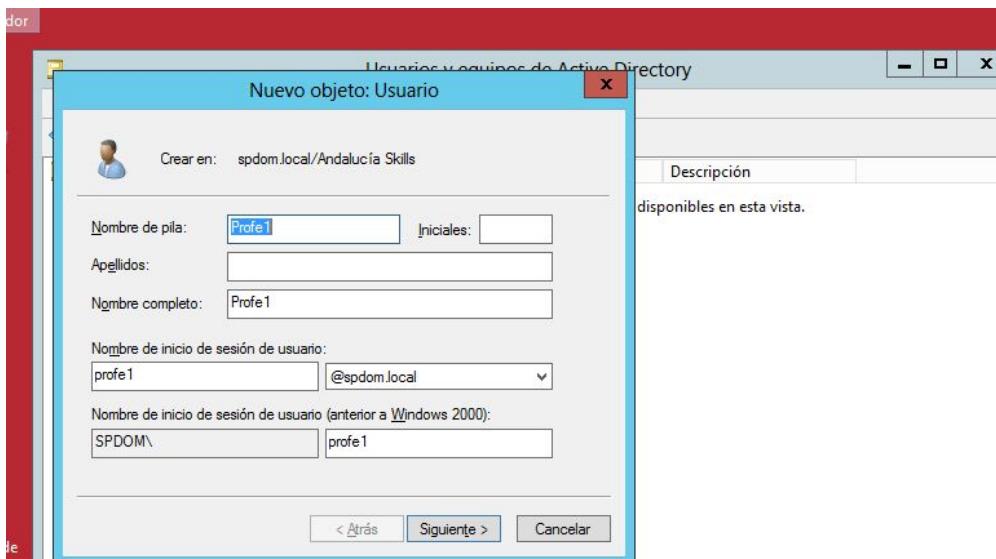
Ahora, vamos a añadir los siguientes usuarios al Directorio Activo:

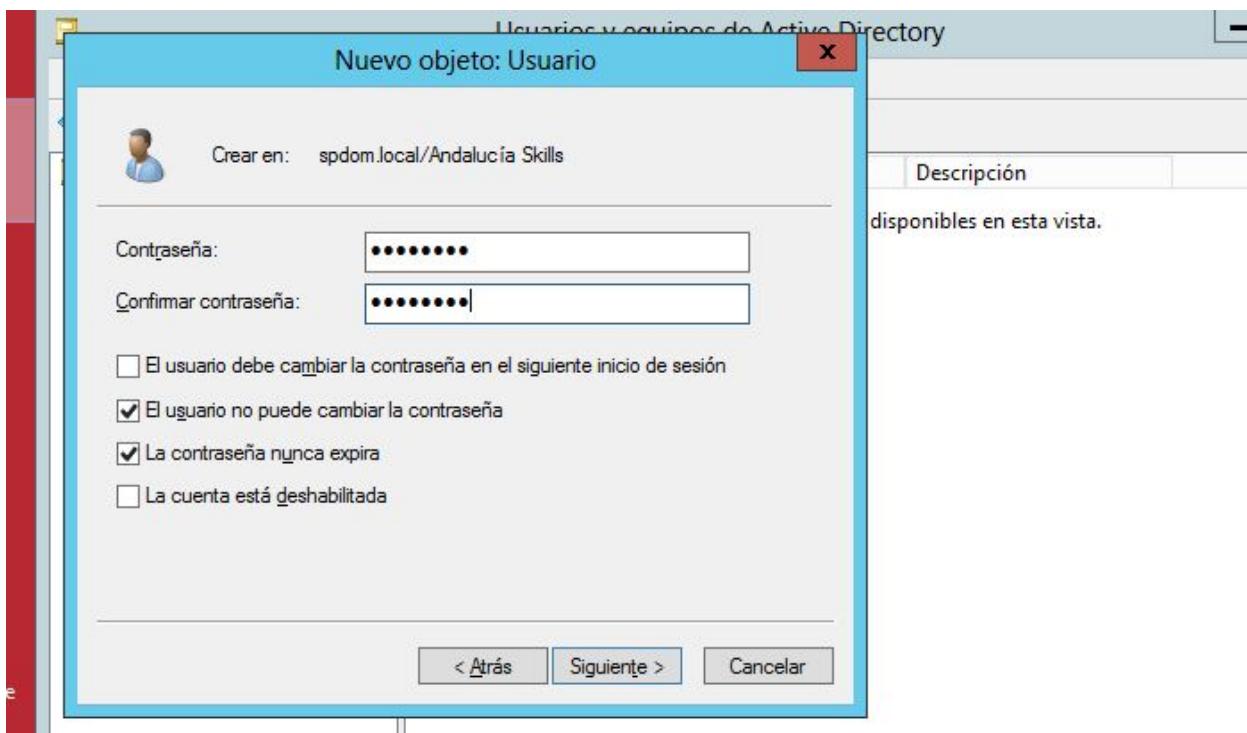
Grupo	Nombre Usuario	Contraseña	Puede cambiar la contraseña	Cambiar contraseña en el primer inicio de sesión
Profesorado	Profel	P@ssw0rd	No	No
Estudiantes	Estul	S@ludit0s	No	No
Administrativos	Ad1	Sp@in2017	No	No

Para ello, nos dirigiremos al panel de Administración de Usuarios y Equipos de Active Directory. Crearemos una unidad Organizativa para tener mayor control sobre los usuarios y los grupos.

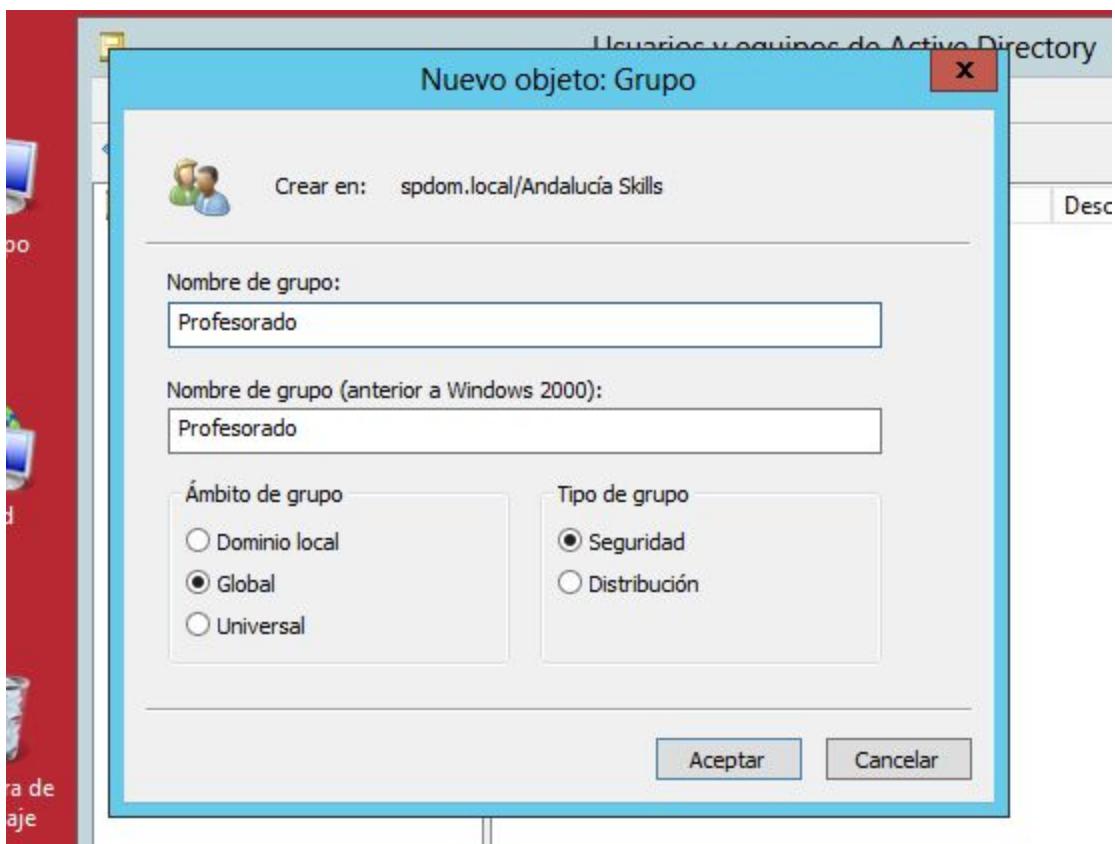


Creamos los usuarios:





Y los grupos:



Ya los tenemos todos creados:

Nombre	Tipo	Descripción
Ad1	Usuario	
Administrativos	Grupo de segu...	
Estu1	Usuario	
Estudiantes	Grupo de segu...	
Profel	Usuario	
Profesorado	Grupo de segu...	

Ahora, añadimos cada usuario a cada grupo correspondiente.

Nombre

Ad1	Copiar...
Administrativos	Agregar a un grupo...
Estu1	...
Estudiantes	...
Profel	...
Profesorado	...

Tipos de objeto...

Ubicaciones...

Comprobar nombres

Selección de grupos

Seleccionar este tipo de objeto: Grupos o Entidades de seguridad integradas

Desde esta ubicación: spdom.local

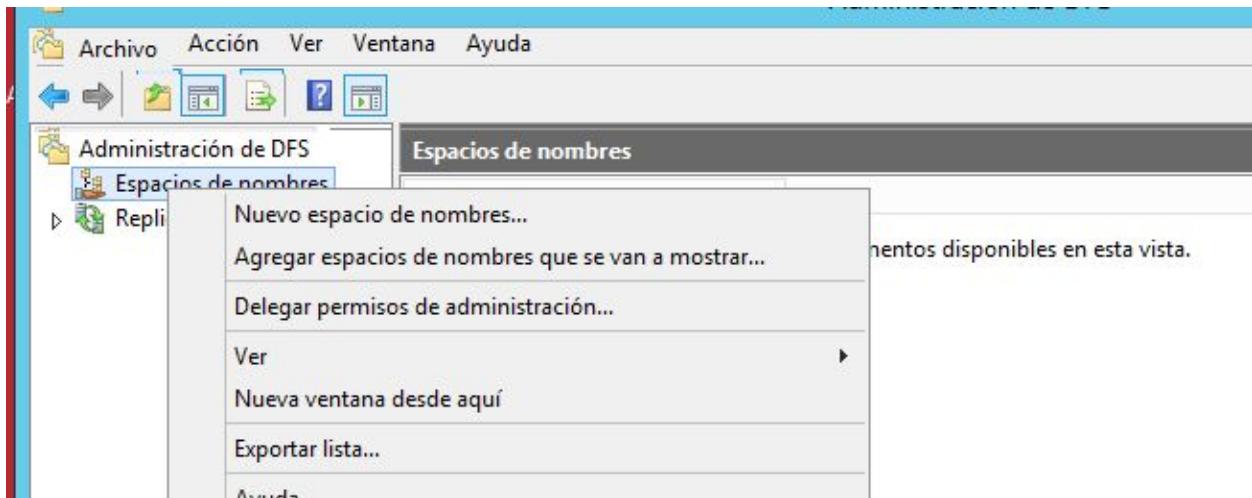
Escriba los nombres de objeto que desea seleccionar ([ejemplos](#)): Estudiantes

Opciones avanzadas...

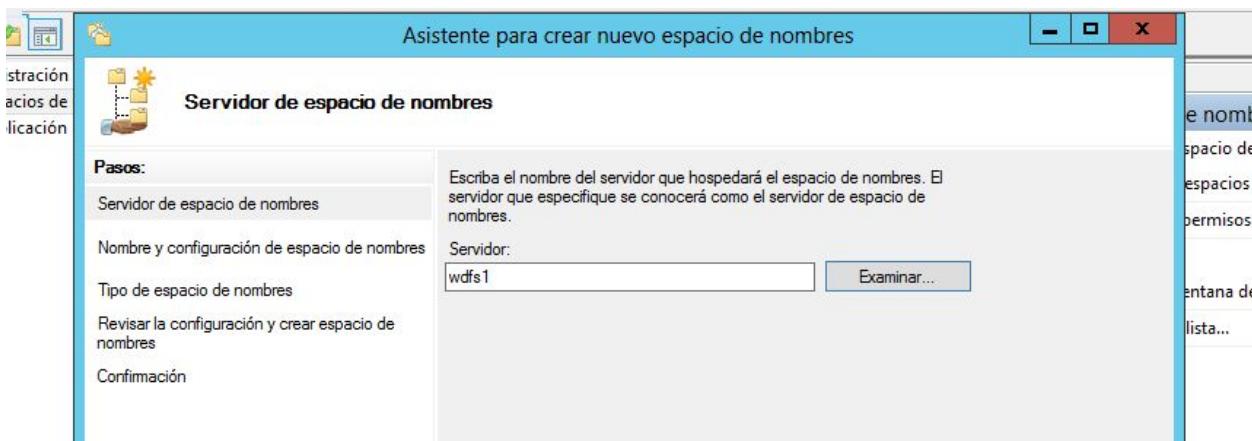
Aceptar Cancelar

5. DFS

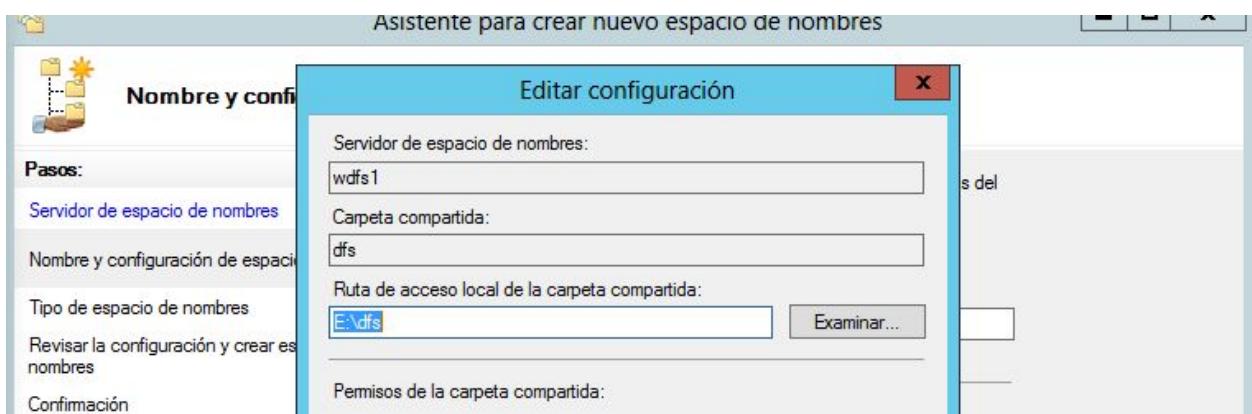
Para ello, nos vamos al panel de administración de DFS y agregamos un nuevo espacio de nombres.



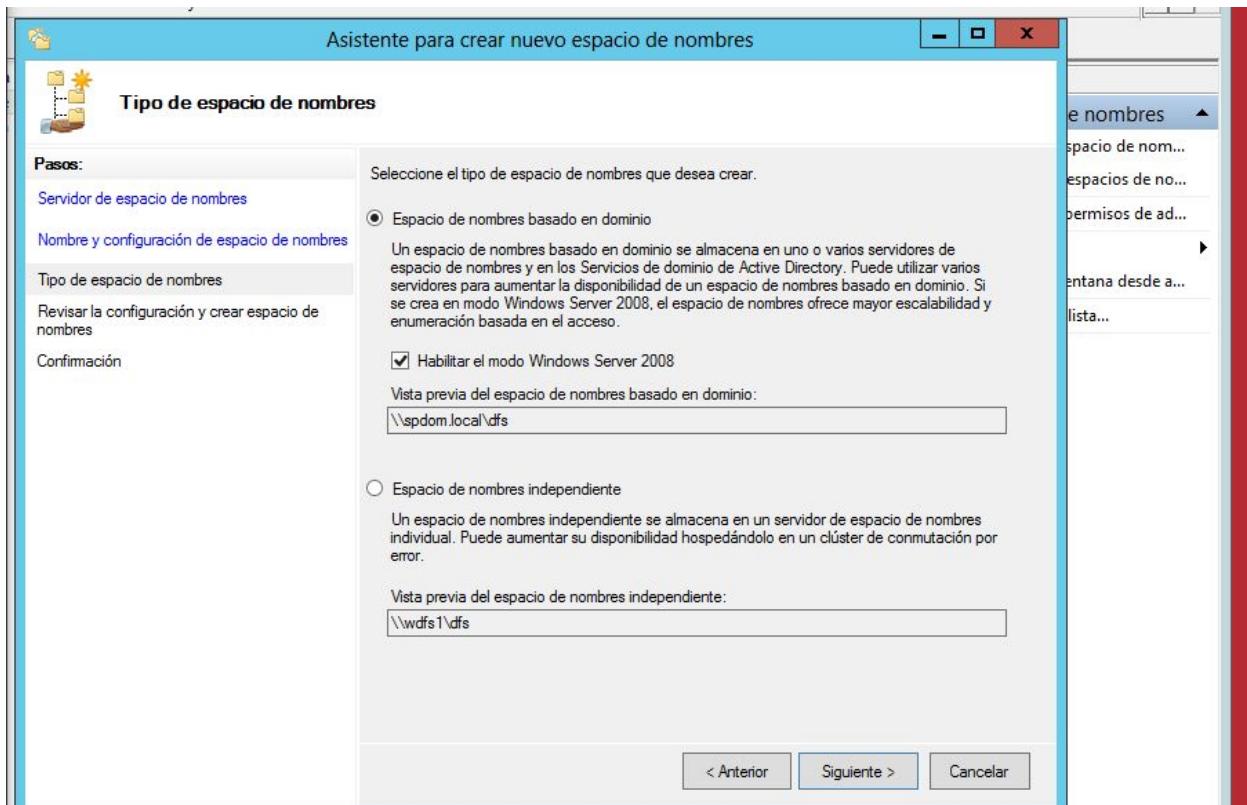
Seleccionamos el Servidor que vamos a usar, en este caso la misma máquina.



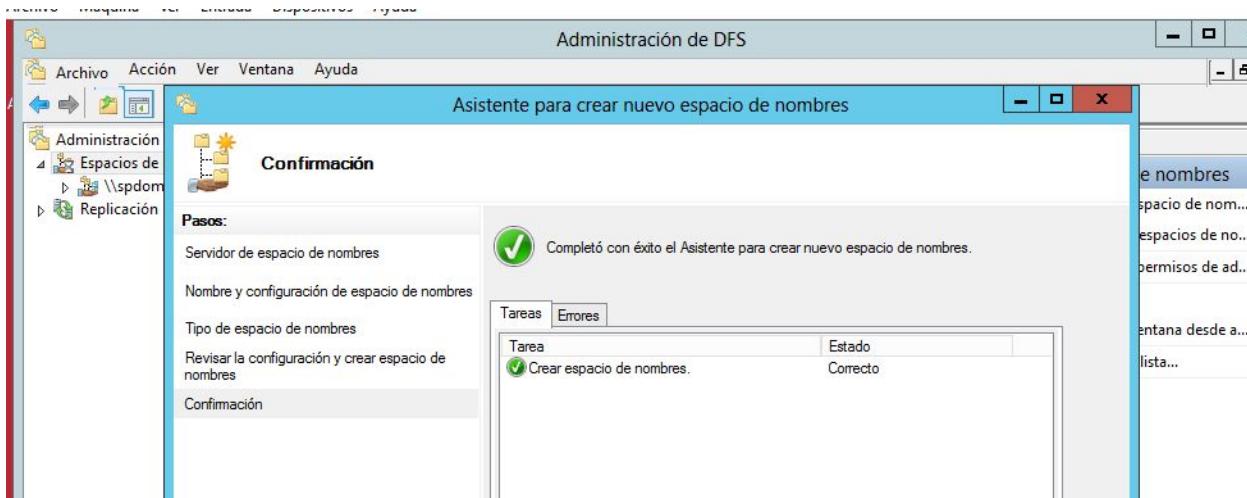
Ahora, elegimos donde vamos a agregar la carpeta, en este caso en el disco E:\.



Continuamos con el asistente:

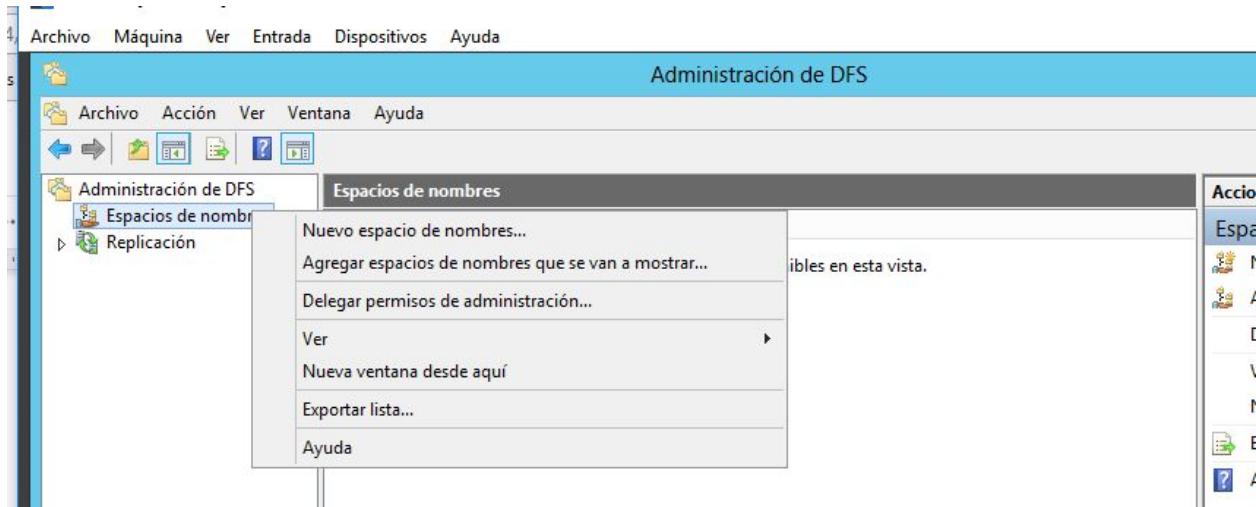


Correcto todo:

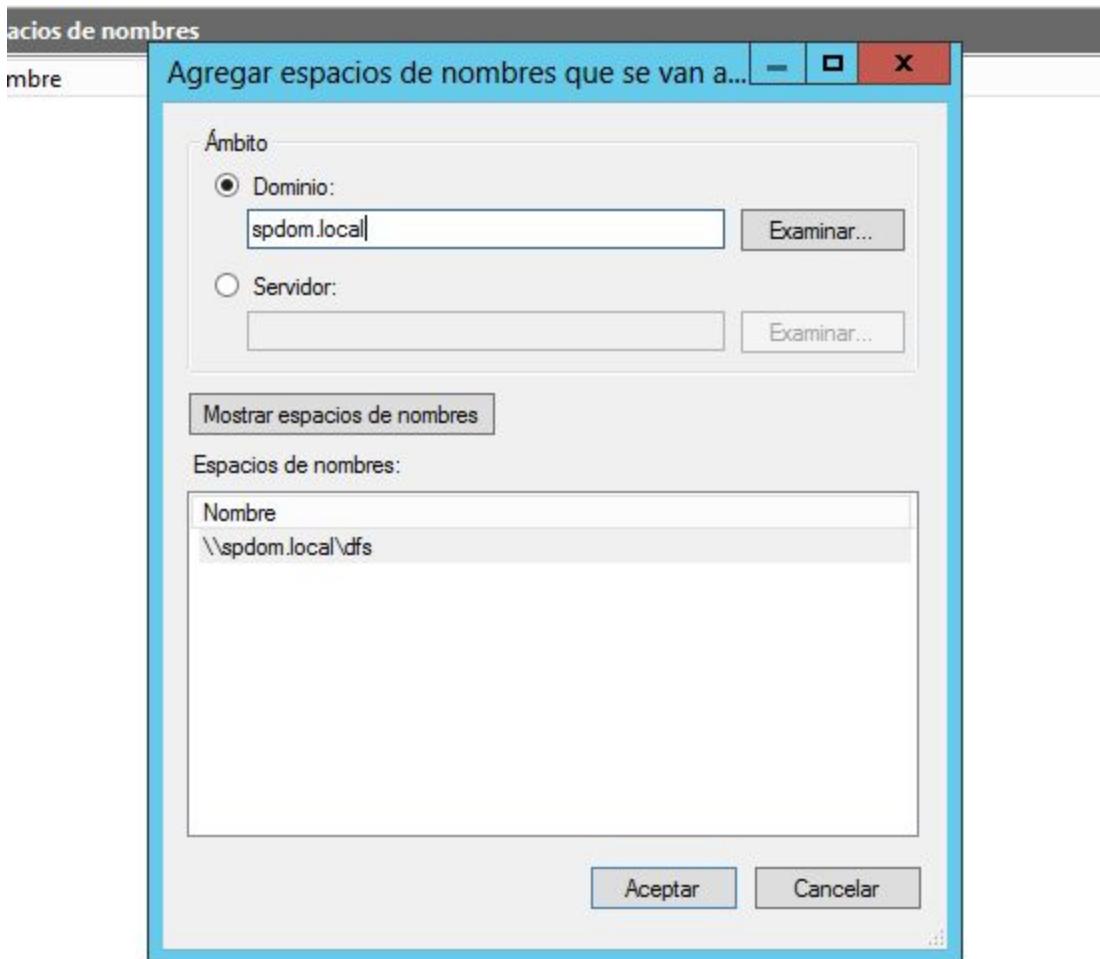


Ahora vamos al siguiente servidor y agregamos este espacio de nombres.

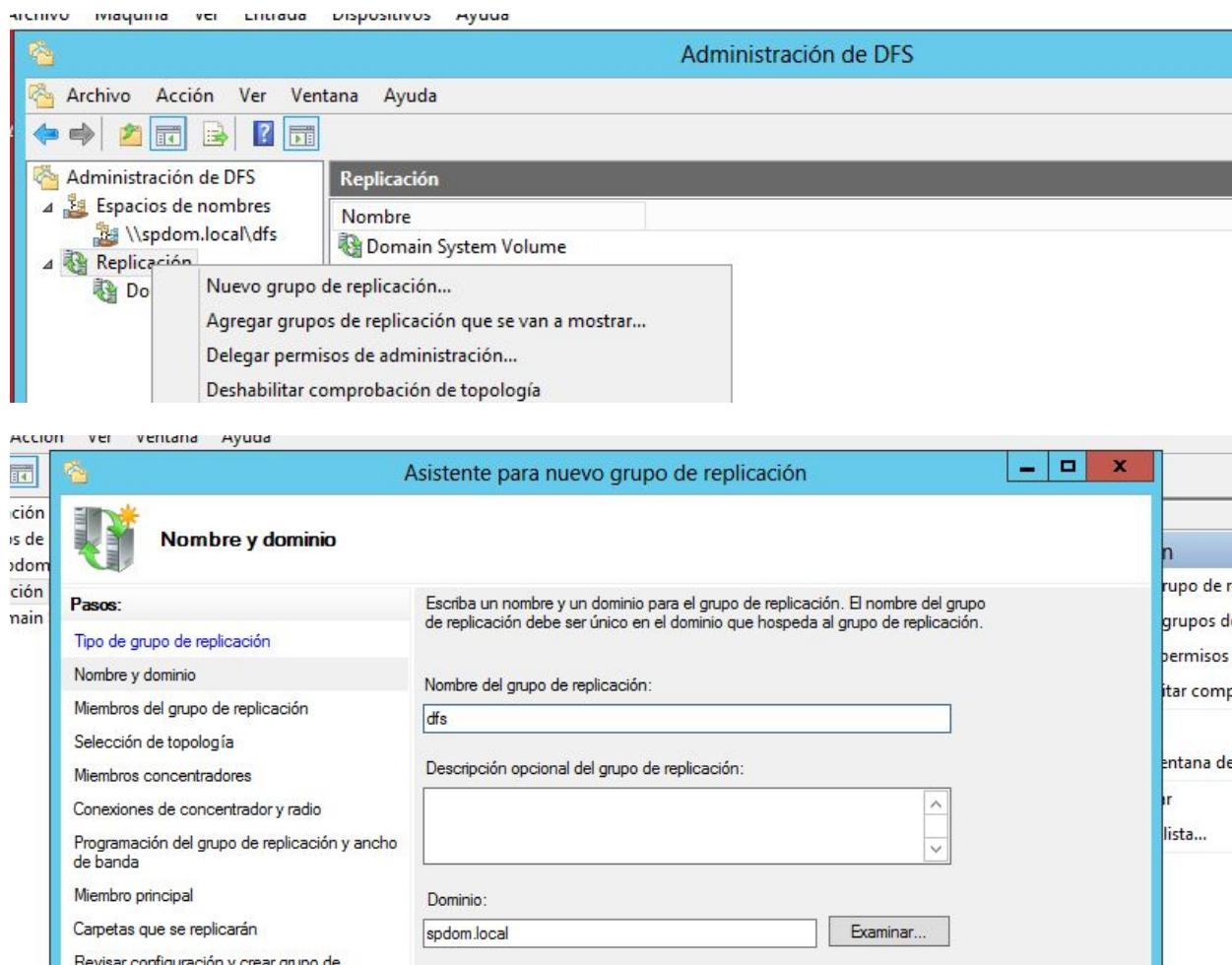
Elegimos la opción Agregar espacios de nombres que se van a mostrar.



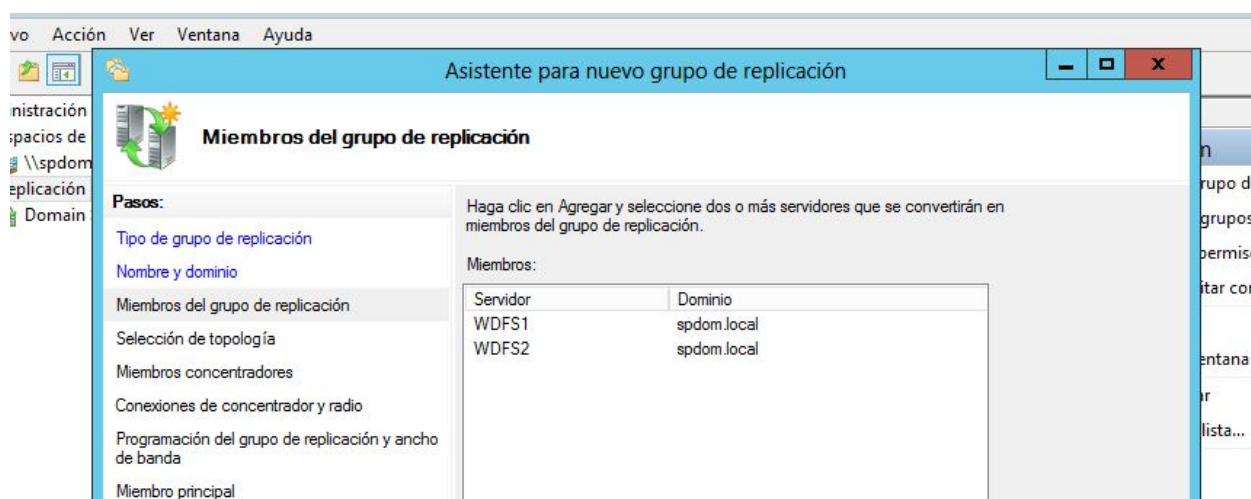
Como la tenemos agregada al dominio ya nos aparece el espacio de nombres creado antes:



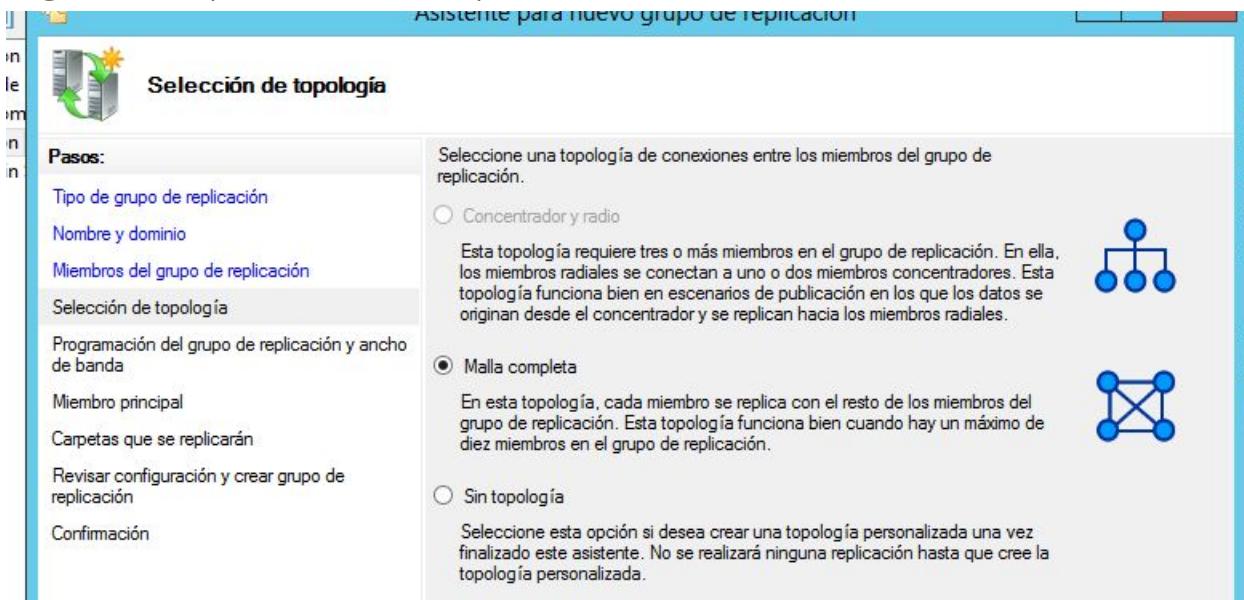
Ahora pasamos a la replicación. Seleccionamos nuevo grupo de Replicación:



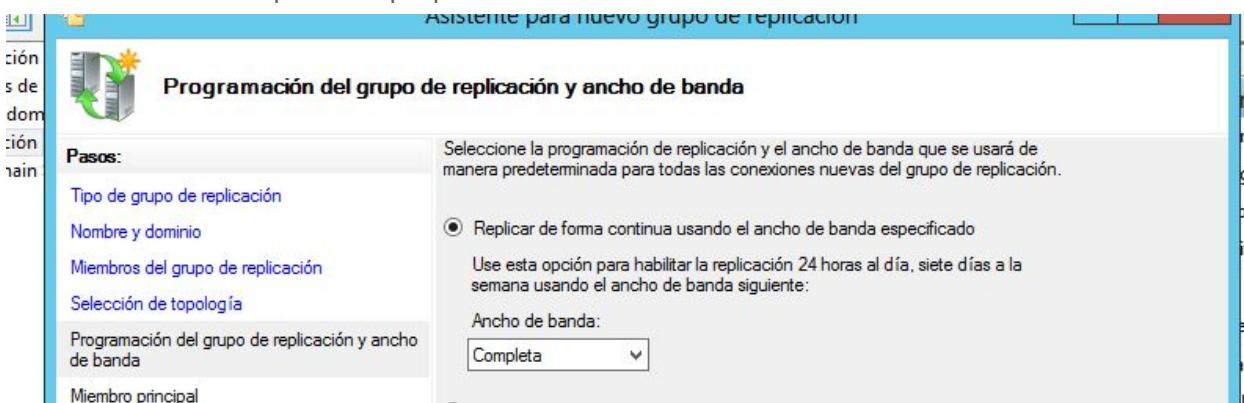
Agregamos los servidores al grupo de replicación:



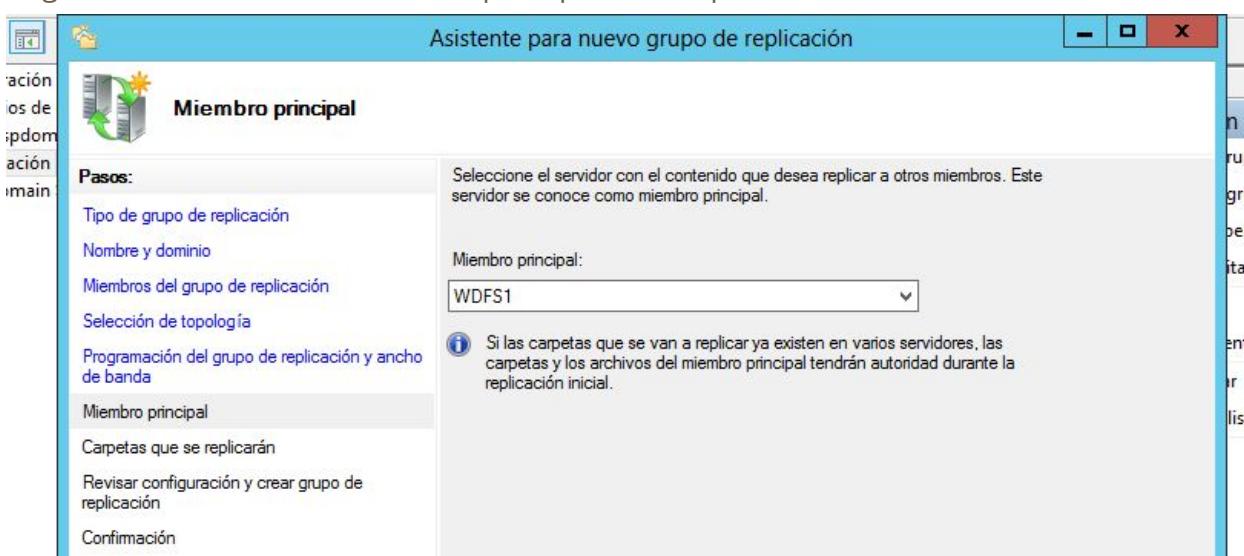
Elegiremos la opción de malla completa:



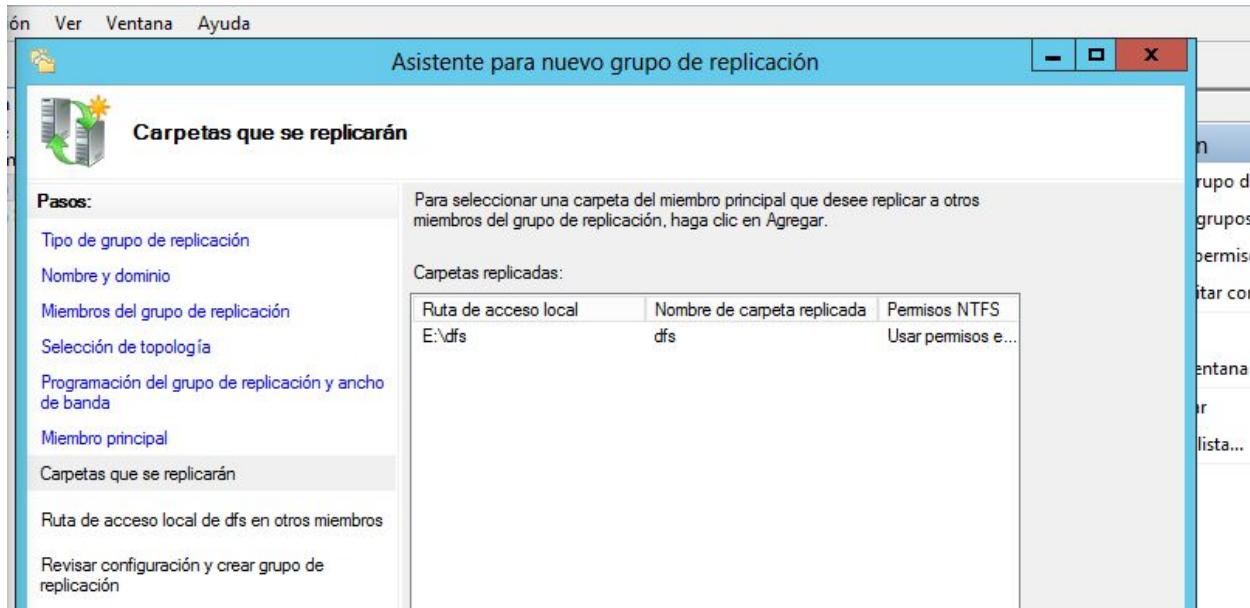
Y seleccionaremos que se replique de forma continua usando todo el ancho de banda:



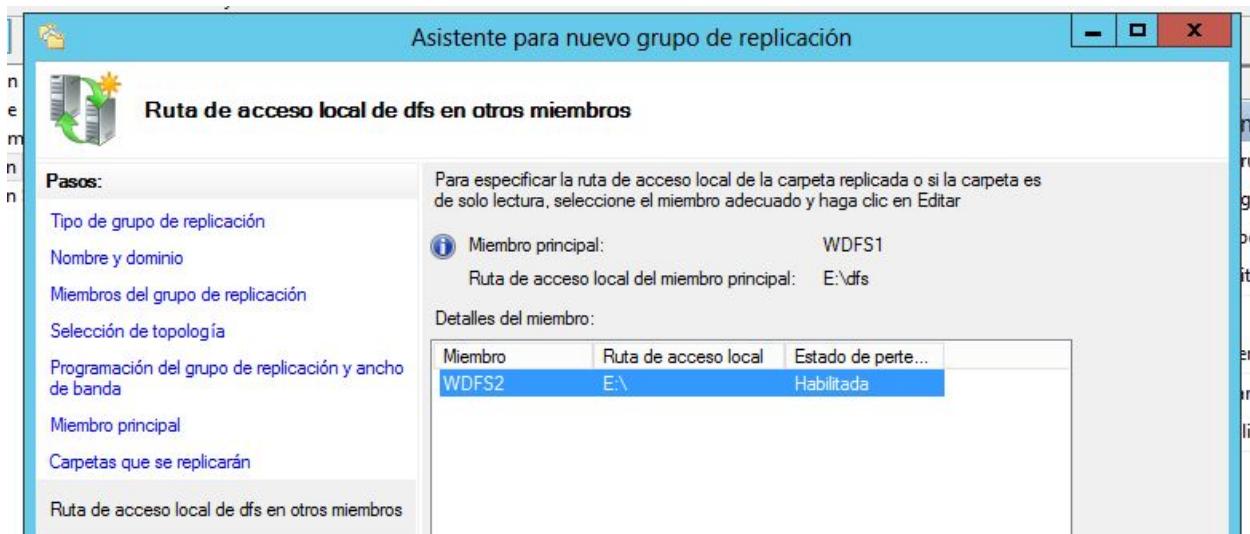
Elegiremos WDFS1 como miembro principal de la replicación:



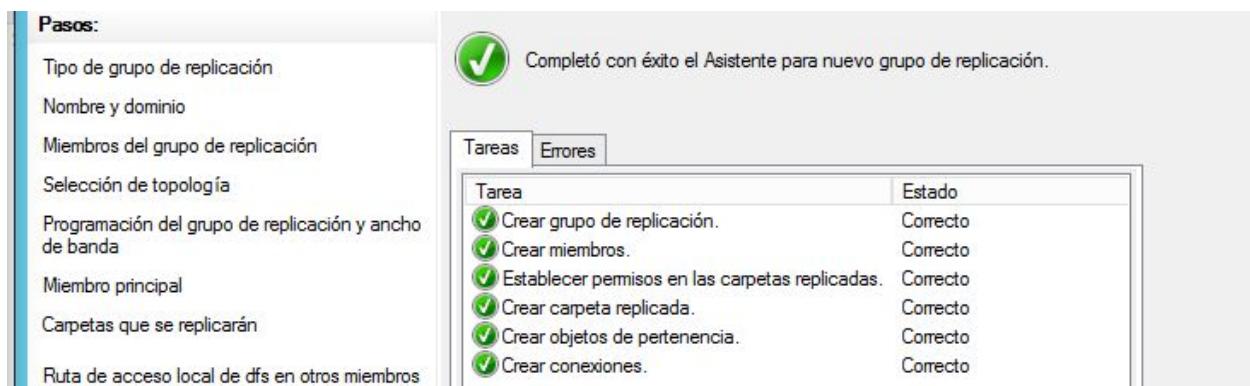
Seleccionaremos la carpeta dfs nuestra como la carpeta a replicar:



Decimos donde guardaremos la carpeta a replicar:

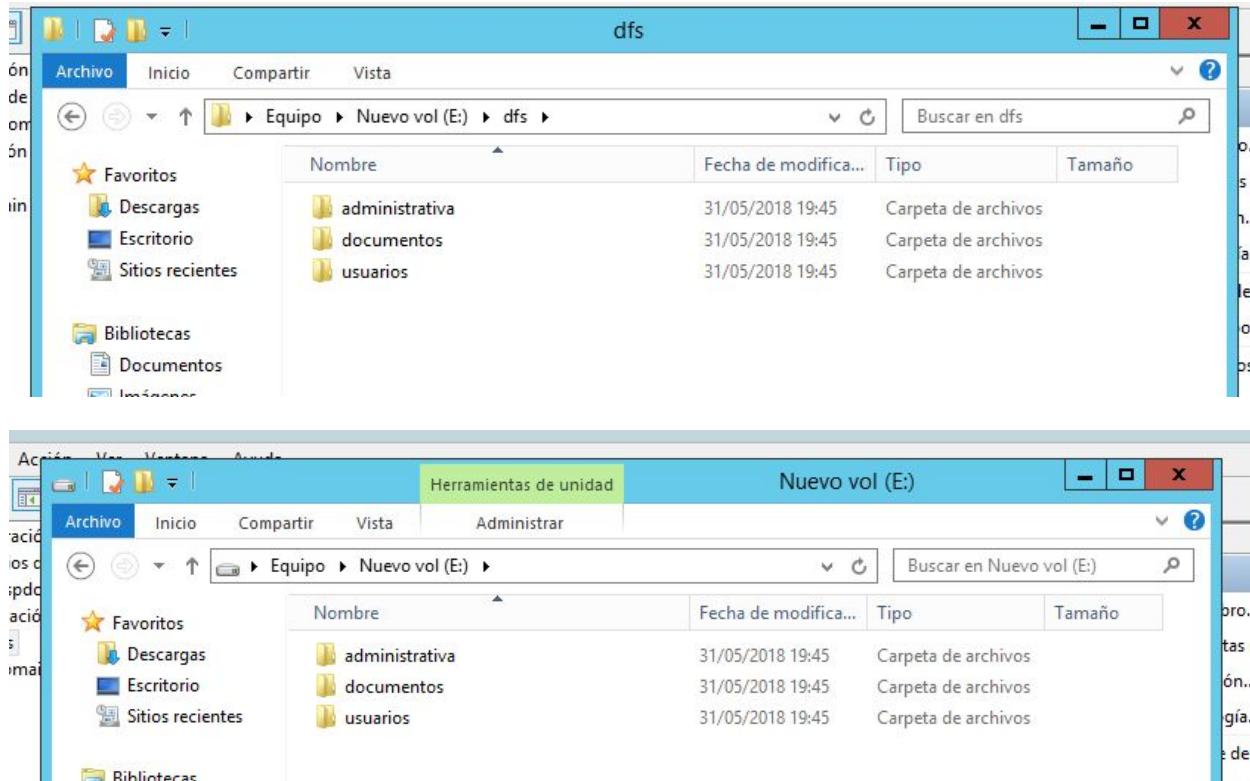


Ahora ya tenemos la replicación preparada y se pondrá a hacerla:

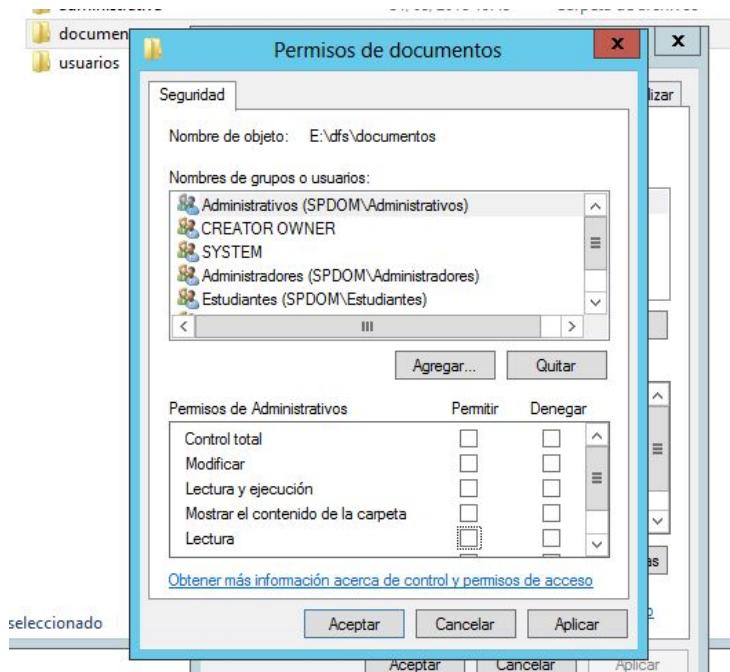


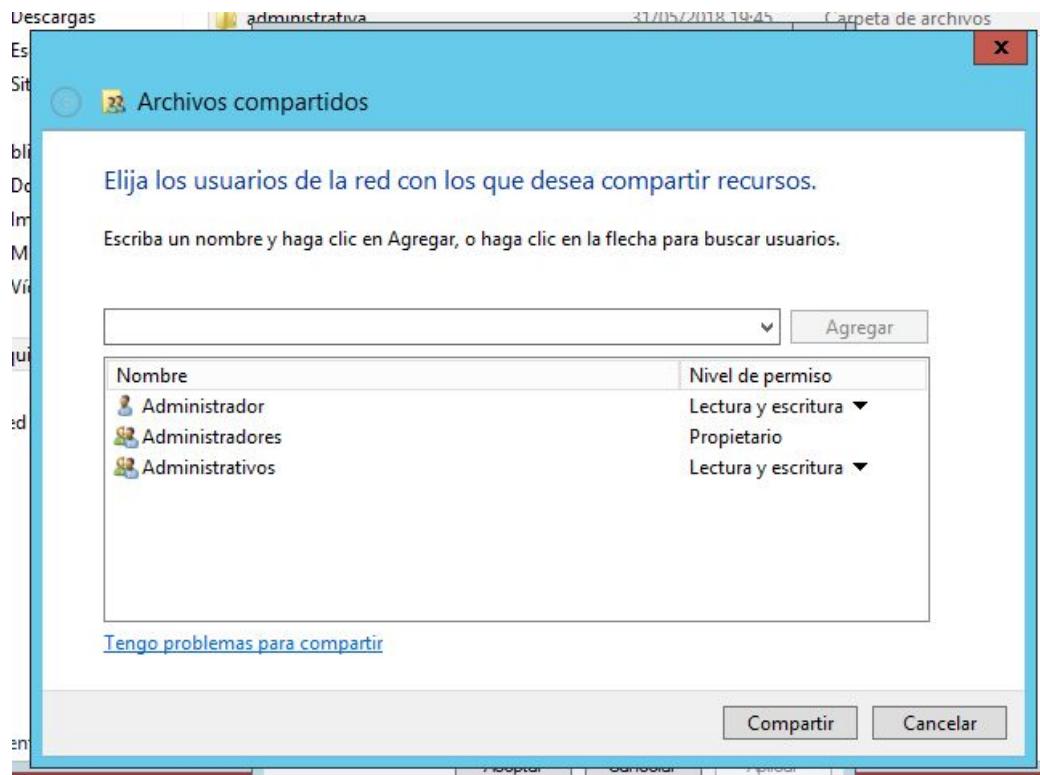
6. Carpetas compartidas y mapeo.

Ahora creamos las carpetas y se crearán en el otro servidor ya que tenemos habilitado el DFS:

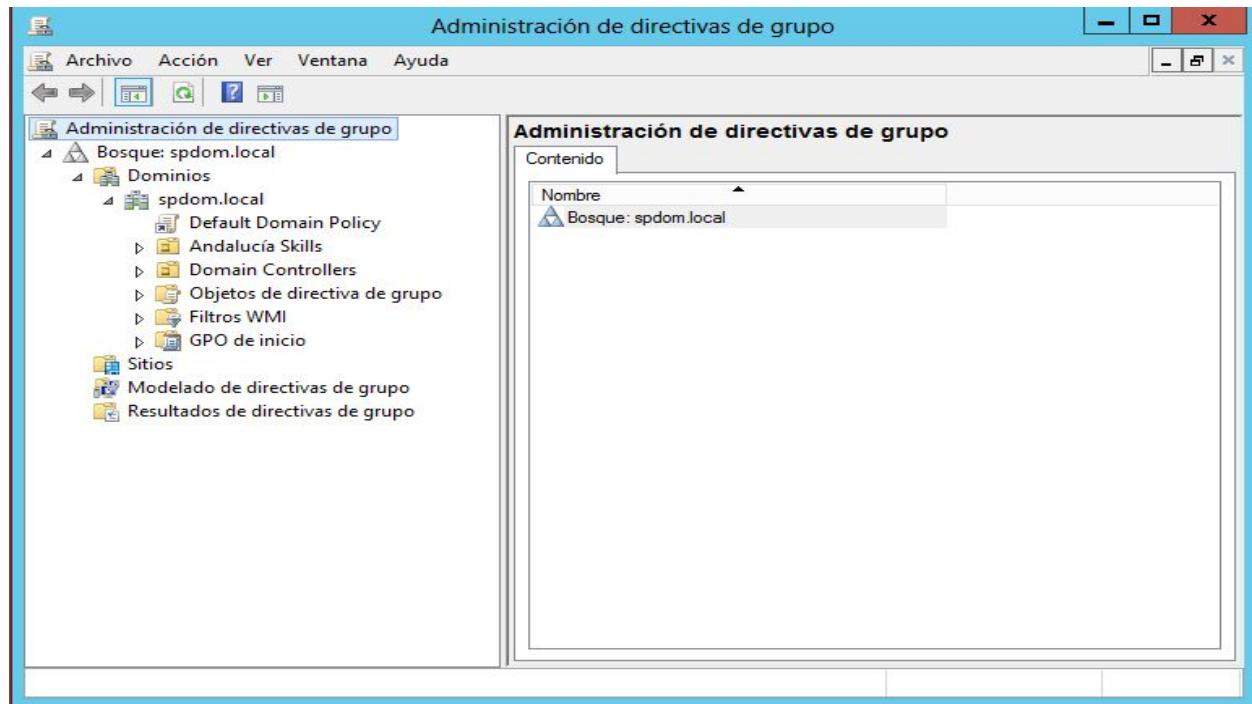


Ahora pasamos a compartir las carpetas y darles los permisos para cada una.

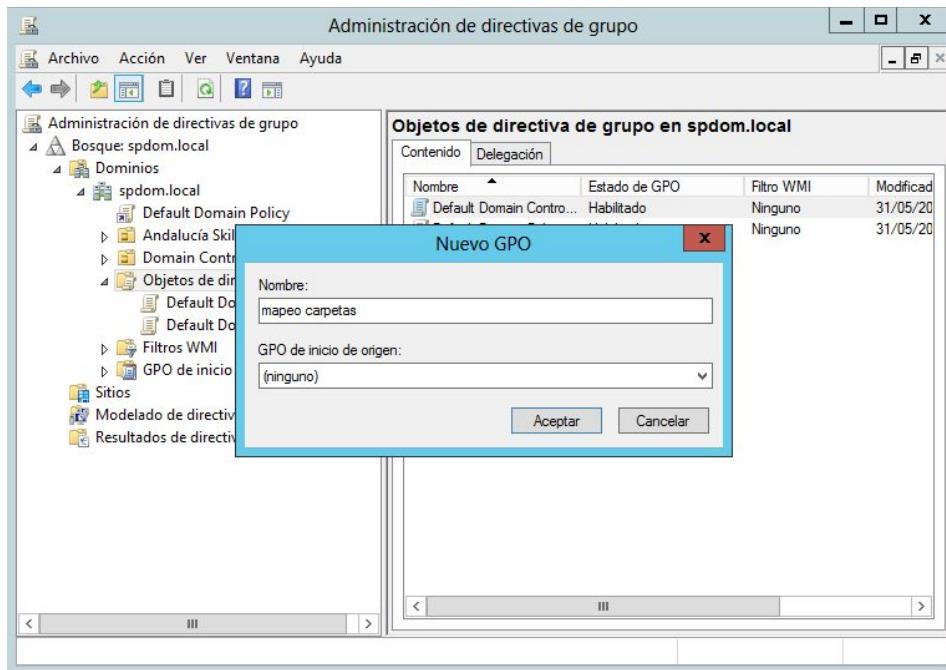




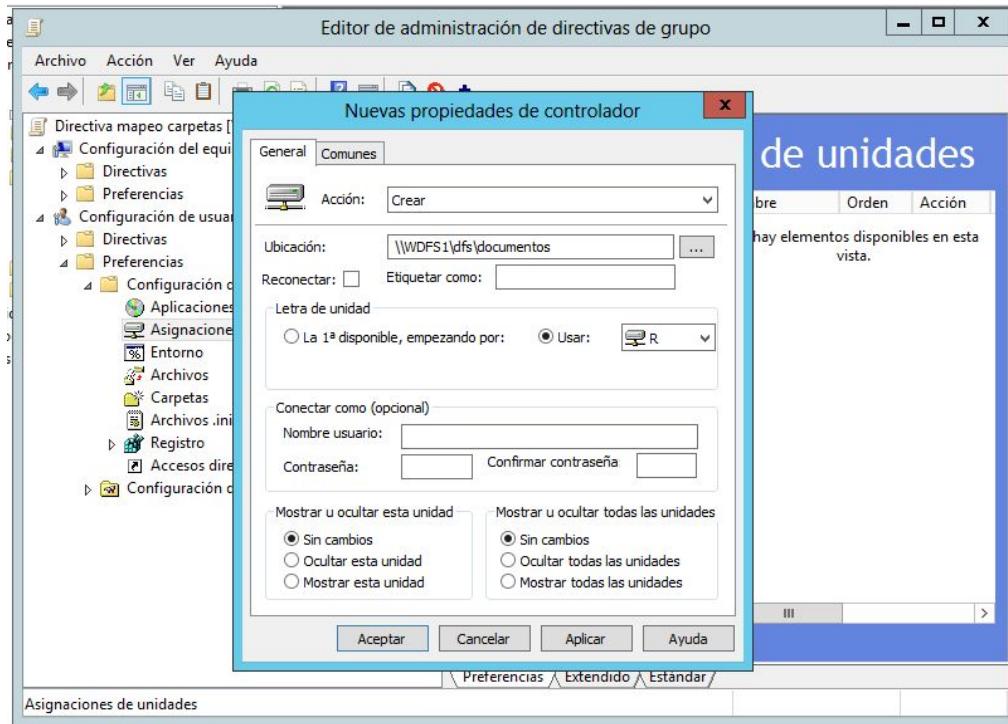
Lo siguiente que vamos a hacer es irnos a el Administrador de Directivas de Grupo para mapear las carpetas en los clientes. Empezaremos con las carpetas de Administradores y Documentos y luego nos centraremos en la de cada usuario.

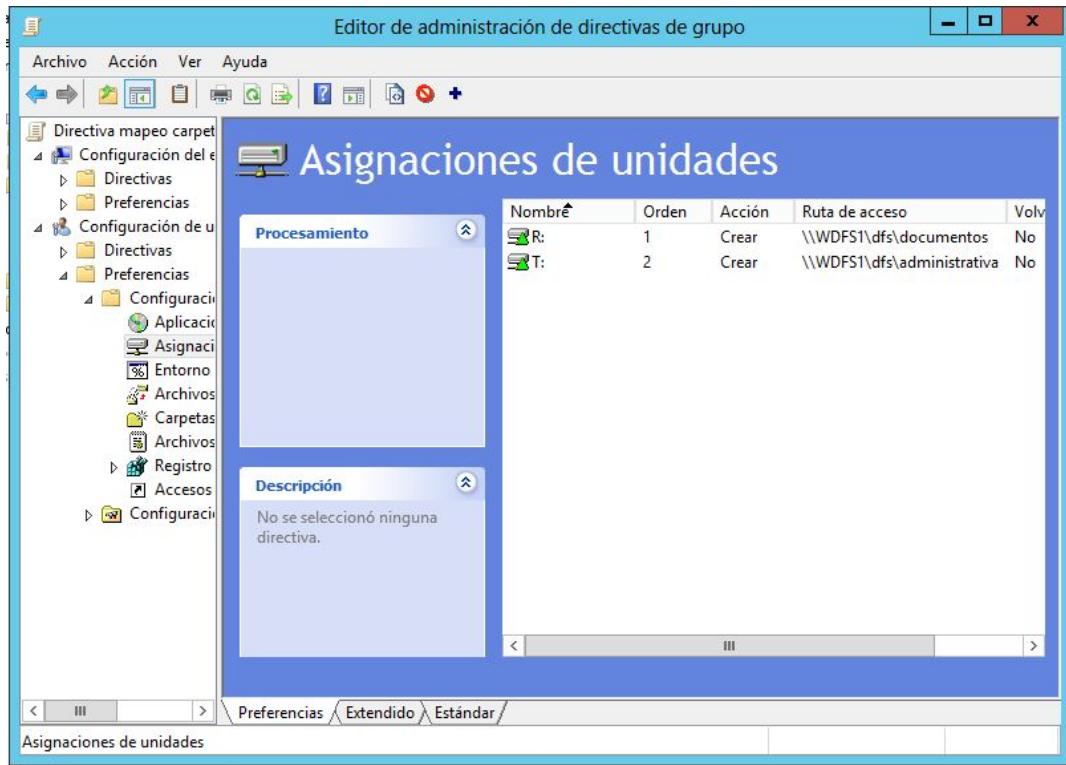


Creamos un nuevo objeto de Política de Grupo:

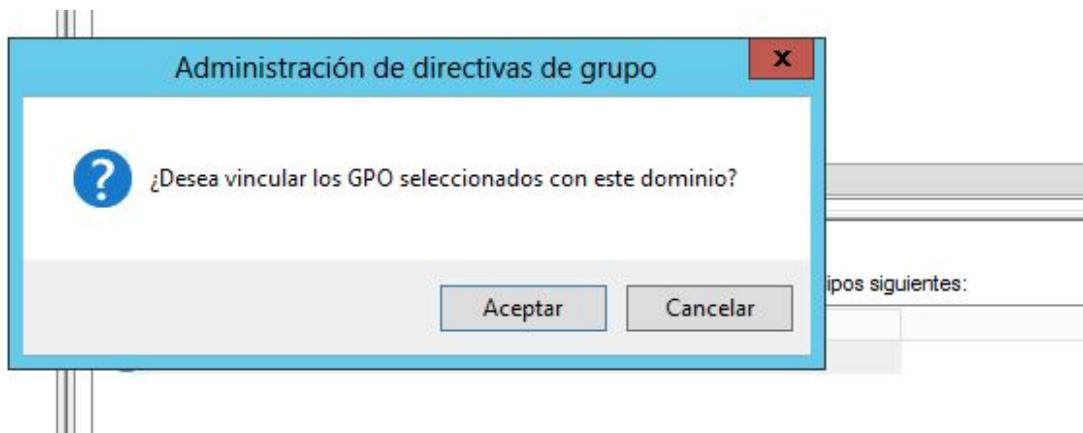


Ahora, editamos el objeto y nos dirigimos a preferencias, configuración de windows y asignaciones de unidades y iremos asignando cada unidad:

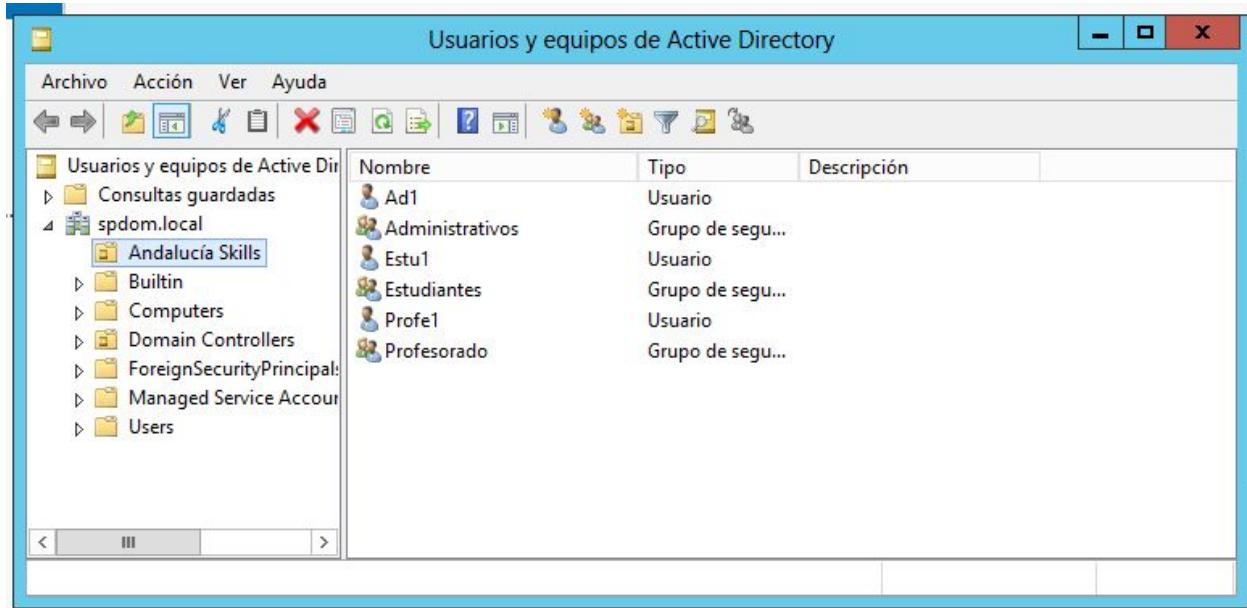




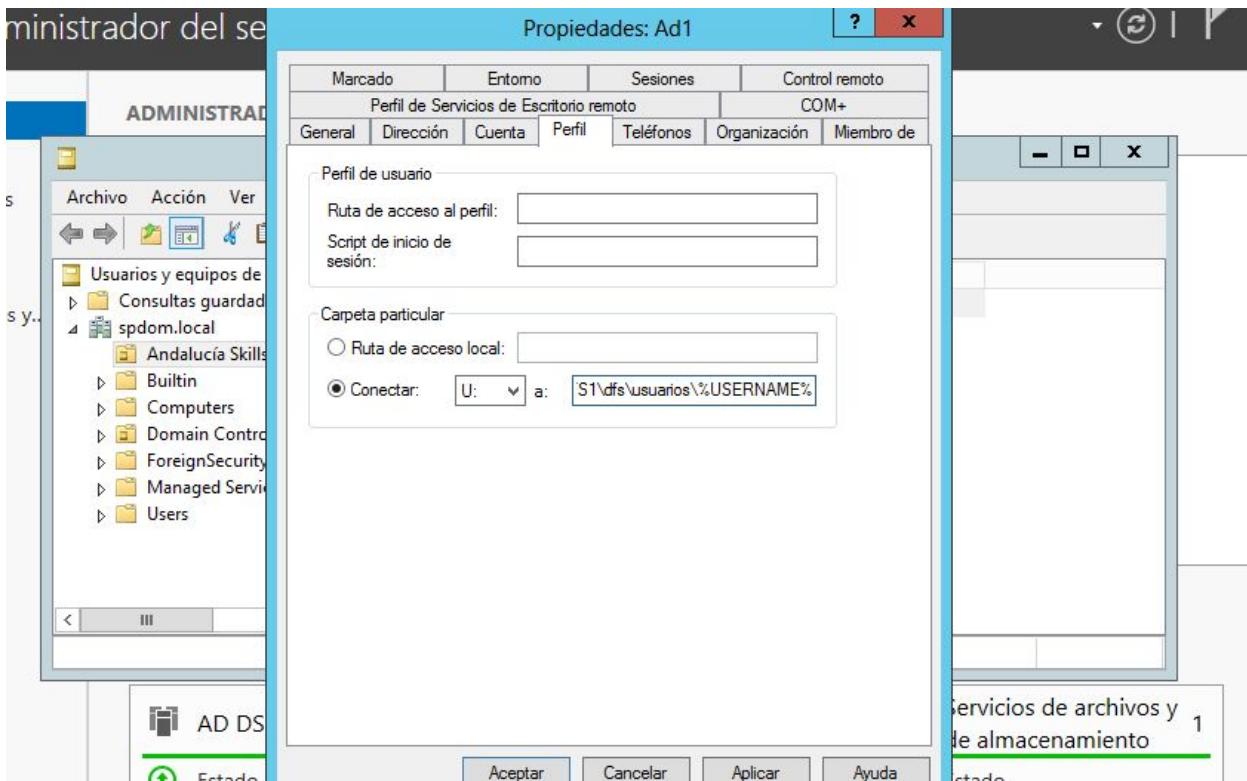
Ahora, ya sólo nos queda aplicar la GPO en nuestro Active Directory.



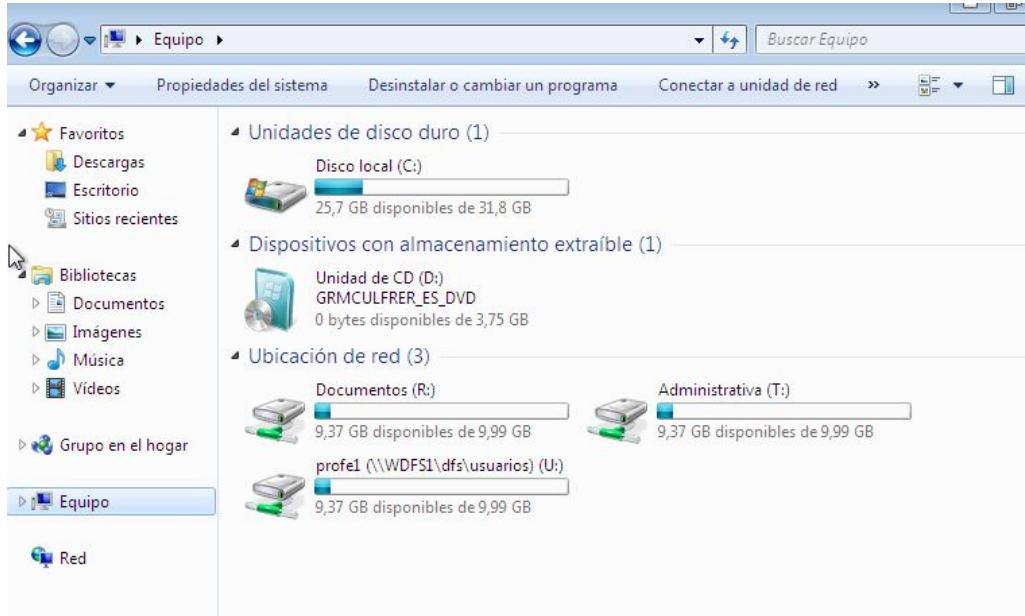
Ahora, pasamos a configurar la de cada usuario. Para ello, nos dirigimos a la configuración de usuarios y equipos del Active Directory.



Ahora, en cada usuario que tengamos deberemos irnos a su configuración y en la pestaña perfil deberemos dar sobre la opción conectar y elegimos la letra en la que queremos mapear (en nuestro caso la letra U). Ahora, deberemos escribir la ruta de la carpeta donde vayamos a crear la carpeta y deberemos añadir %USERNAME%. Quedaría de la siguiente forma:



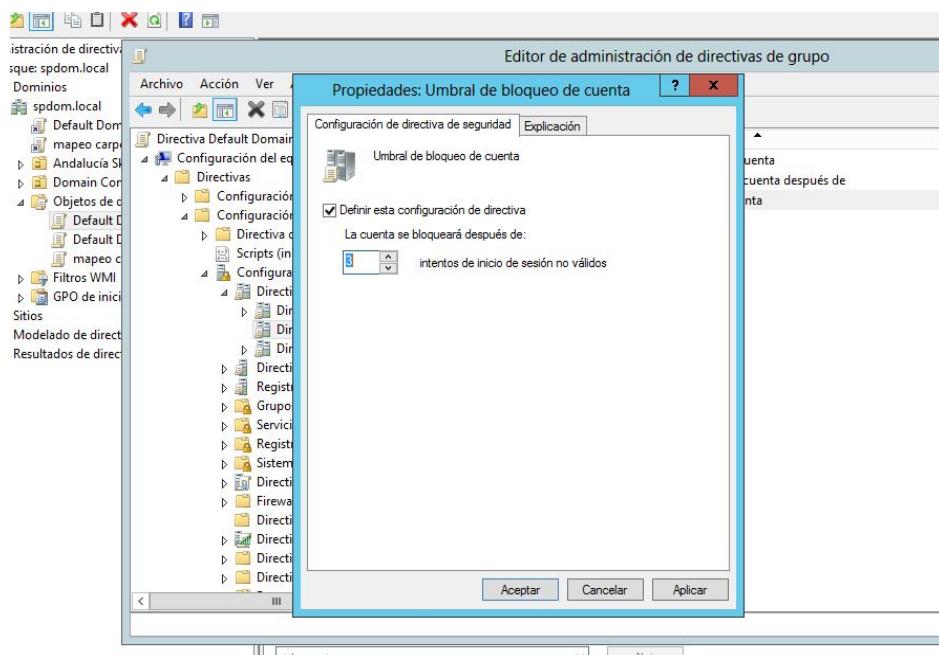
Ahora, aplicamos y guardamos y listo, cuando iniciemos sesión con el usuario nos mostrará las carpetas mapeadas:

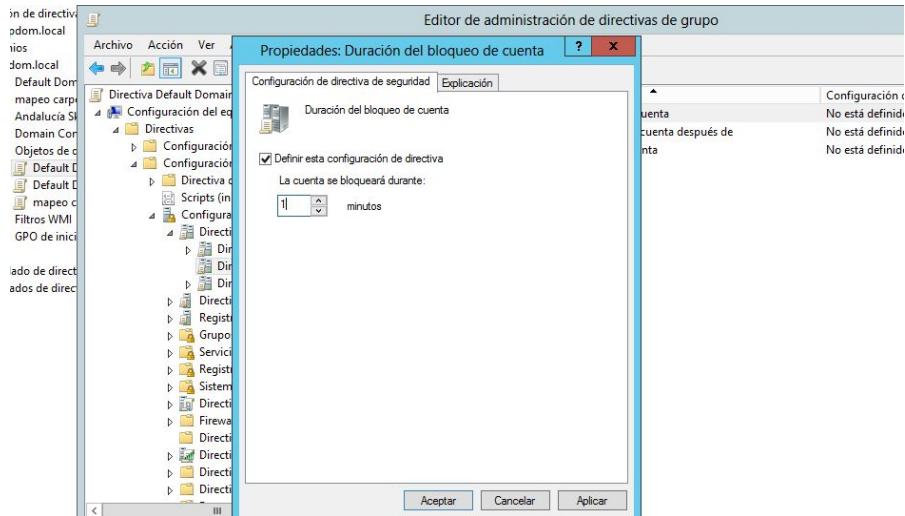


7. Políticas de dominio

Pasamos a la configuración de las políticas de dominio.

Vamos a hacer que cuando un usuario falle con su contraseña 3 veces se le bloquee la cuenta durante un minuto. Para ello nos vamos al panel de Políticas de grupo y nos dirigiremos a **Configuración del equipo\Configuración de Windows\Configuración de seguridad\Directivas Cuenta\directiva de bloqueo**. Ahí deberemos cambiar los valores y habilitarlo.

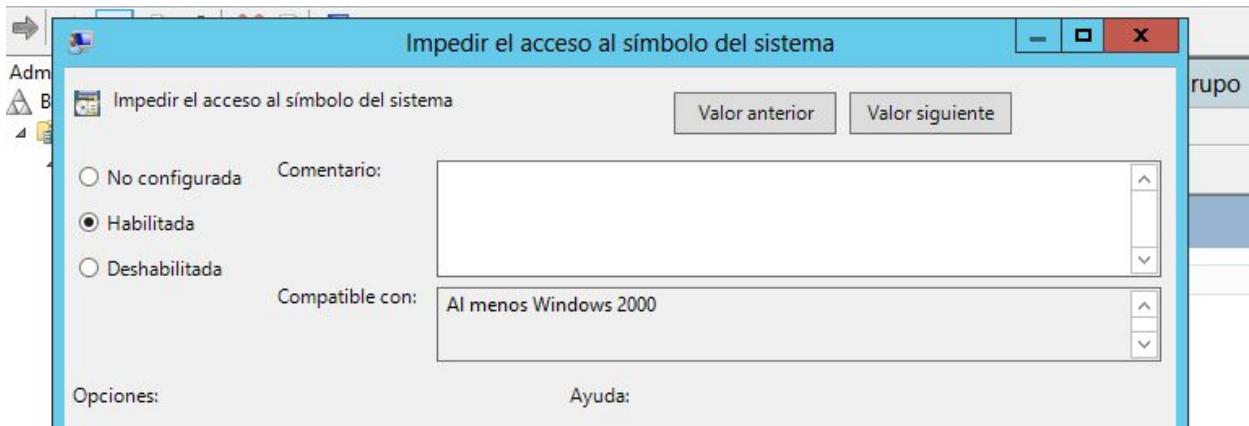




Ahora, pasamos a la política de grupo de que el grupo Estudiantes no pueda ejecutar el comando cmd. Para ello, crearemos una política nueva y añadiremos al grupo para que la use.

The screenshot shows the 'Nuevo GPO' (New GPO) dialog box being used to create a new Group Policy Object (GPO) named 'usuarios cmd'. The 'Filtrado de seguridad' (Security Filtering) section is visible in the background, showing that the 'Estudiantes' group is selected as the target audience for the GPO.

Ahora, editamos la política y comenzamos a configurarla. Nos dirigimos a **Configuración del usuario\Directivas\Plantillas Administrativas...\Sistema\Impedir el acceso al símbolo del sistema**

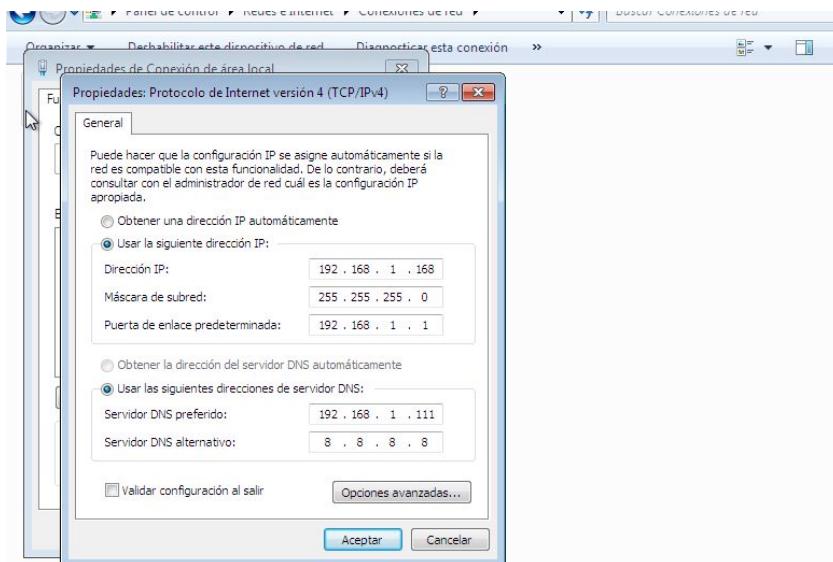


Con esto, habremos terminado.

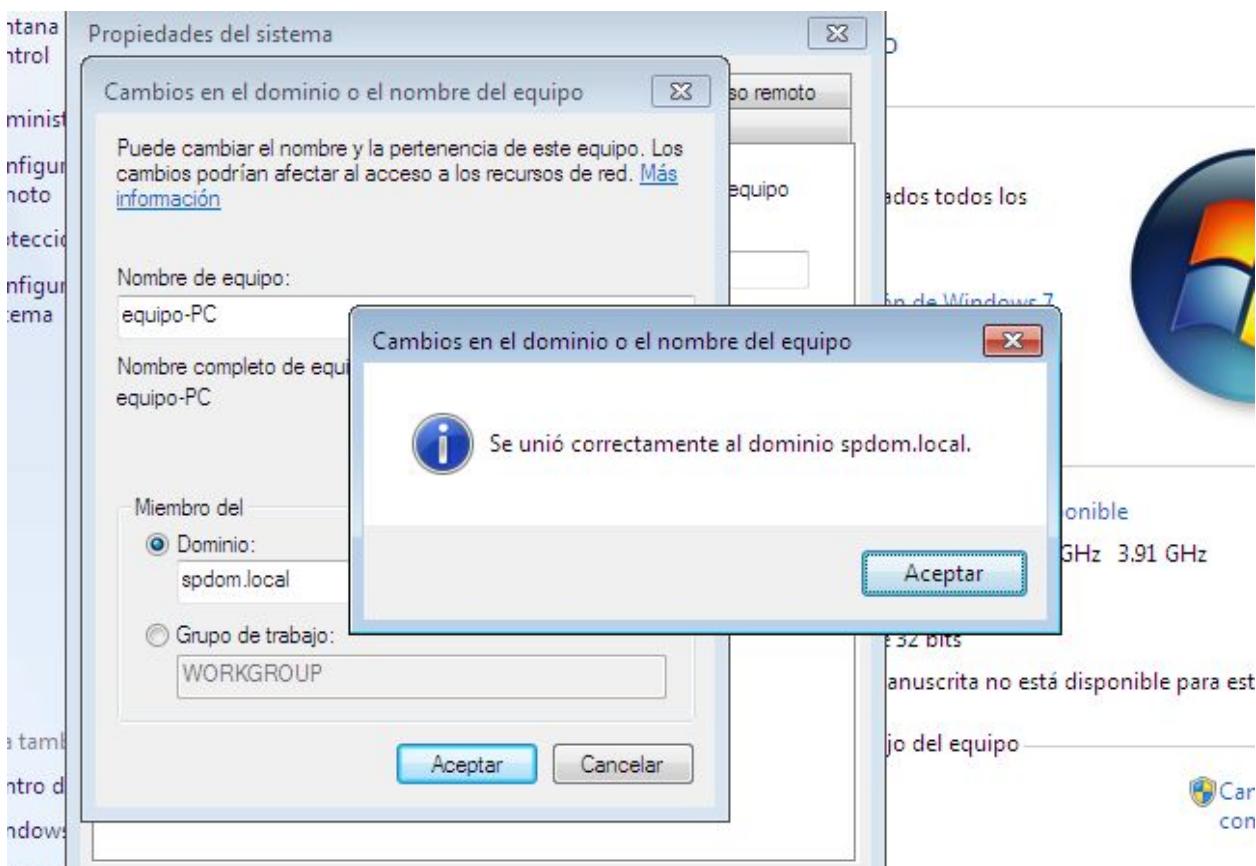
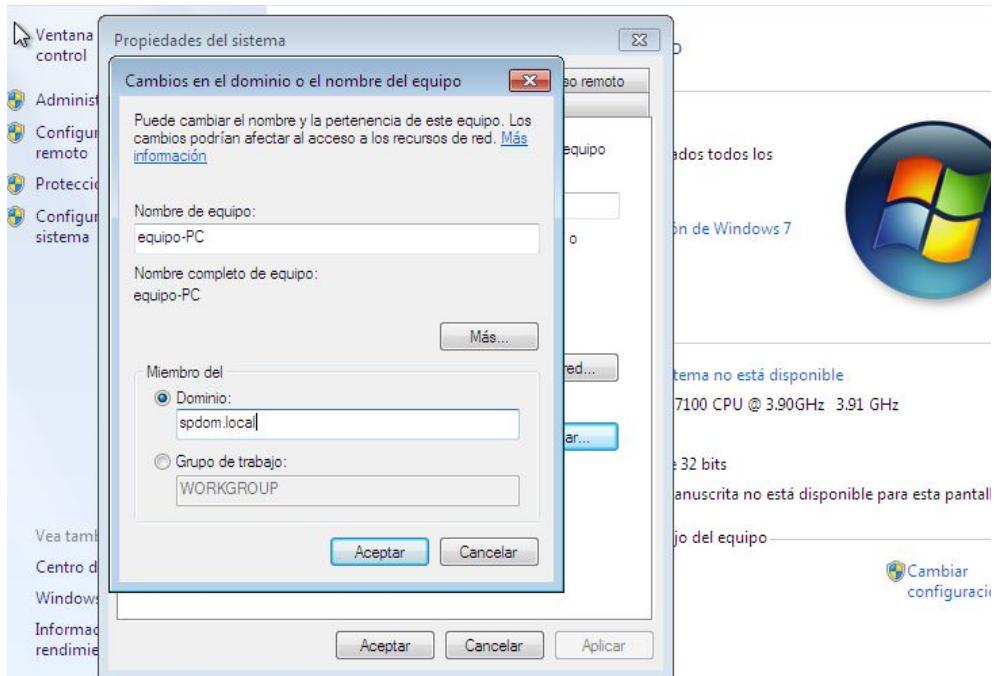
8. Configuración de los clientes de Directorio Activo

- Windows

Para ello. Lo primero que vamos a hacer será asignar la dirección de IP y las DNS de nuestros servidores correspondientes a nuestro cliente windows. Para ello, nos dirigimos a la configuración de la tarjeta de red:



Ahora, nos dirigimos a Sistema y agregamos el equipo al dominio



- Ubuntu

Lo primero que debemos hacer será instalar lo siguiente en nuestro Ubuntu:

```
$ sudo apt install krb5-user samba sssd ntp
```

El siguiente paso, configurar Kerberos. Para ello editamos el archivo de configuración **/etc/krb5.conf** y especificamos:

```
[libdefaults]
default_realm = SPDOM.LOCAL
ticket_lifetime = 24h #
renew_lifetime = 7d
```

Sólo con esto el sistema debería ser capaz de encontrar nuestros controladores de dominio.

Configuramos NTP para que sincronice la hora del controlador de dominio primario, agregando a **/etc/ntp.conf** la siguiente línea:

```
server dc1.calnus.com
```

Lo siguiente será configurar Samba:

Editamos **/etc/samba/smb.conf** y añadimos lo siguiente:

```
[global]
workgroup = SPDOM
client signing = yes
client use spnego = yes
kerberos method = secrets and keytab
realm = SPDOM.LOCAL
security = ads
```

Ahora, toca configurar sssd. Para ello, nos vamos al fichero **/etc/sssd/sssd.conf**:

```
[sssd]
services = nss, pam
config_file_version = 2
domains = SPDOM.LOCAL

[domain/SPDOM.LOCAL]
id_provider = ad
access_provider = ad

# Use this if users are being logged in at /.
```

```
# This example specifies /home/DOMAIN-FQDN/user as $HOME. Use with
pam_mkhomedir.so
override_homedir = /home/%d/%u
```

Ahora es necesario establecer los permisos y propietario del archivo. Si no están correctos, SSSD se negará a arrancar:

```
sudo chown root:root /etc/sssd/sssd.conf
sudo chmod 600 /etc/sssd/sssd.conf
```

También editaremos el /etc/hosts y agregaremos a este archivo el FQDN de la máquina Linux que vas a unir al dominio.

```
192.168.1.111 wdfs1 wdfs1.spdom.local
```

Ahora, pasamos a unirnos al Active Directory.

Primero, reiniciamos los servicios NTP y Samba:

```
$ sudo systemctl restart ntp
$ sudo systemctl restart smbd nmbd
```

Ahora, solicitamos un ticket de Kerberos a nuestro Administrador de dominio

```
$ sudo kinit Administrador
Password for Administrador@SPDOM.LOCAL:
```

Para unirnos al dominio usamos el siguiente comando:

```
sudo net ads join -k
```

9. Script de creación de usuarios.

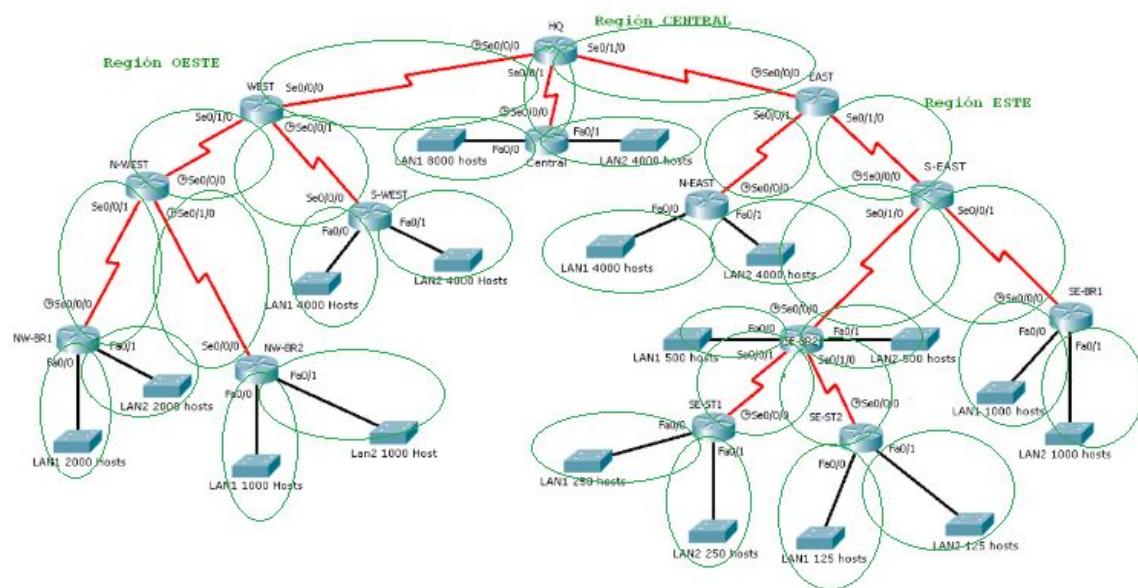
```
for ($i = 1; $i -lt 99; $i++)
{
    New-ADUser -SamAccountName "Estu$i" -Name "Estu$i" -GivenName "Estu$i" ` 
    -Surname "Estudiante" -DisplayName "Estu$i" -Path 'DC=spdom,DC=local'
}
```

4. VLSM

En esta última prueba, como hemos mencionado anteriormente, vamos a hacer subnetting usando la Red 172.16.0.0/16 de forma que tenemos que aprovechar de forma más eficiente todas las direcciones ip para cada red del escenario atendiendo al número de host de cada dispositivo.

Lo primero que debemos hacer, es identificar las redes del escenario y asignarle la máscara de red que van a usar cada una.

Aquí tenemos las redes identificadas:



En total tenemos unas 31 redes de las cuales habrá:

- 1 con máscara /19
- 5 con máscara /20
- 2 con máscara /21
- 4 con máscara /22
- 2 con máscara /23
- 2 con máscara /24
- 2 con máscara /25
- 13 con máscara /30

Por lo tanto ya sólo nos queda hacer subnetting de la dirección de red 172.16.0.0/16:

Nº IP's útiles	Dirección de red	Máscara
8190	172.16.0.0/19	255.255.224.0
4094	172.16.32.0	255.255.240.0
4094	172.16.48.0	255.255.240.0
4094	172.16.64.0	255.255.240.0
4094	172.16.80.0	255.255.240.0
4094	172.16.96.0	255.255.240.0
2046	172.16.112.0	255.255.248.0
2046	172.16.120.0	255.255.248.0
1022	172.16.128.0	255.255.252.0
1022	172.16.132.0	255.255.252.0
1022	172.16.136.0	255.255.252.0
1022	172.16.140.0	255.255.252.0
510	172.16.144.0	255.255.254.0
510	172.16.146.0	255.255.254.0
254	172.16.148.0	255.255.255.0
254	172.16.149.0	255.255.255.0
125	172.16.150.0	255.255.255.128
125	172.16.150.128	255.255.255.128
2	172.16.151.0	255.255.255.252
2	172.16.151.4	255.255.255.252
2	172.16.151.8	255.255.255.252
2	172.16.151.12	255.255.255.252
2	172.16.151.16	255.255.255.252
2	172.16.151.20	255.255.255.252
2	172.16.151.22	255.255.255.252
2	172.16.151.26	255.255.255.252

2	172.16.151.30	255.255.255.252
2	172.16.151.34	255.255.255.252
2	172.16.151.38	255.255.255.252

Con esto, terminamos la prueba de VLSM y finalizamos el proyecto.