

# P2-CONMUTADORES

Alejandro Rodríguez Rojas  
Luis Vázquez Alejo  
Miguel Córdoba Aranda  
Manuel Durán Muñoz

# Índice

1 Introducción.....	3
Captura de tráfico.....	3
STP.....	3
2 Conectándose al Switch.....	3
3 Cambio de IP.....	4
4 Cambio de IP Maquinas.....	5
5 Captura de Tráfico.....	6
6 STP.....	7
7 Conclusión.....	10

# 1 Introducción

Vamos a realizar el siguiente ejercicio:

## Captura de tráfico

1. Averigua como configurar el commutador para capturar el tráfico de un puerto determinado.
2. Configúralo para que un ordenador conectado al puerto 20 pueda analizar el tráfico del puerto 1.
3. Realiza una conexión entre dos ordenadores conectados al puerto 1 y 2 y captura el tráfico desde el puerto 20.

## STP

1. Conecta dos commutadores entre sí utilizando un enlace conectado al puerto 1 de cada uno.
2. Conecta un ordenador a cada commutador utilizando el puerto 10.
3. Configura las direcciones de cada ordenador para que sean accesibles. Intenta enviar un ping de uno a otro y después a la dirección de difusión (broadcast)
4. Agrega un enlace entre el puerto 2 de los dos commutadores.
5. Averigua qué enlaces están activos y cuales no. ¿Qué commutador está configurado como la raíz del árbol de expansión?
6. Apaga y reinicia los dos commutadores para comprobar si cambia la asignación.
7. Configura ambos commutadores para que la raíz del árbol de expansión sea siempre el commutador 2.

# 2 Conectándose al Switch

Debemos enchufar el Switch en la red eléctrica y presionar el botón de Factory Default(Altamente recomendado usar un clip para volver a los valores de fábrica).

En este caso estaremos utilizando el Switch marca Netgear de modelo FS750T2.

Una vez realizado estas tareas, la IP de configuración del Switch será la 192.168.0.239.

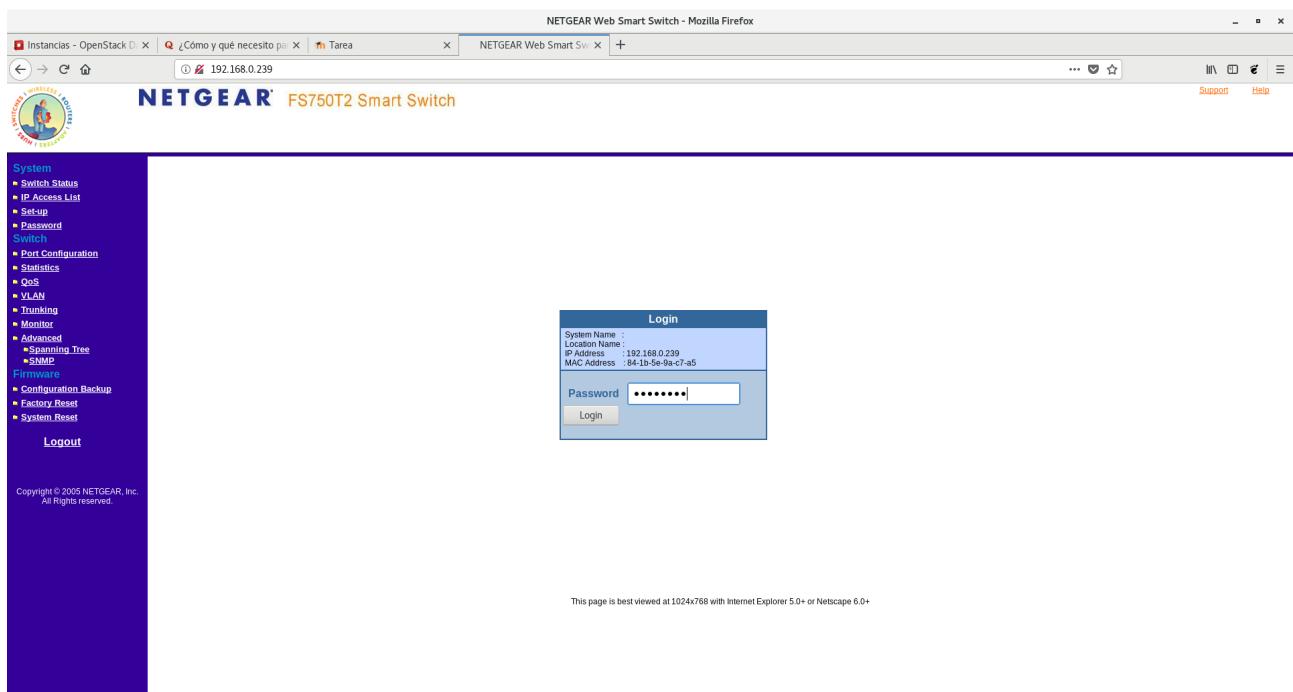
Para conectarnos a la interfaz, tendremos que añadir el rango de IP en nuestra máquina.

Para ello utilizaremos el comando:

ip addr add {IP} dev {Interfaz}

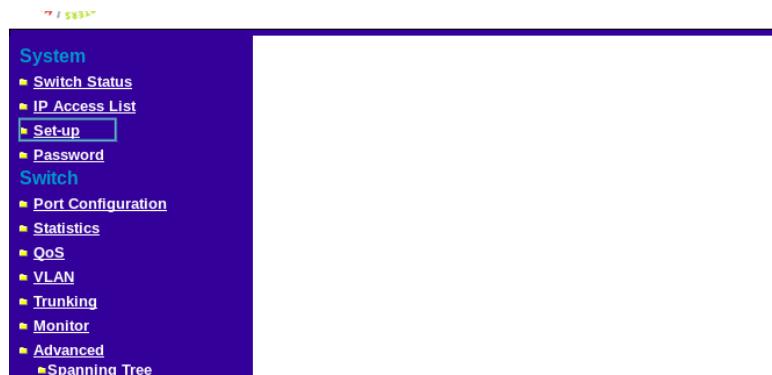
```
alexrr@pc-alex:~$ sudo ip addr add 192.168.0.200/24 dev enp7s0
[sudo] password for alexrr:
alexrr@pc-alex:~$ ping 192.168.0.239
PING 192.168.0.239 (192.168.0.239) 56(84) bytes of data.
64 bytes from 192.168.0.239: icmp_seq=1 ttl=255 time=9.73 ms
64 bytes from 192.168.0.239: icmp_seq=2 ttl=255 time=6.34 ms
64 bytes from 192.168.0.239: icmp_seq=3 ttl=255 time=68.8 ms
64 bytes from 192.168.0.239: icmp_seq=4 ttl=255 time=6.34 ms
```

\*Muy recomendado apagar network-manager antes de realizar el ejercicio  
/etc/init.d/network-manager stop

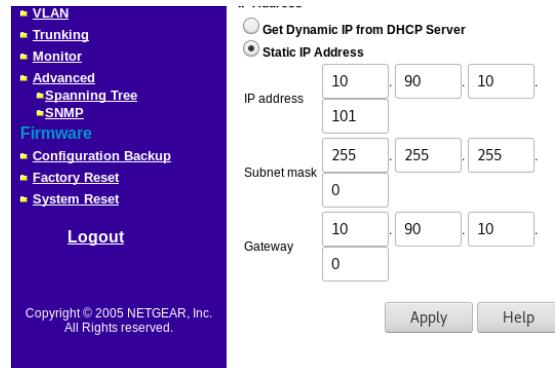


### 3 Cambio de IP

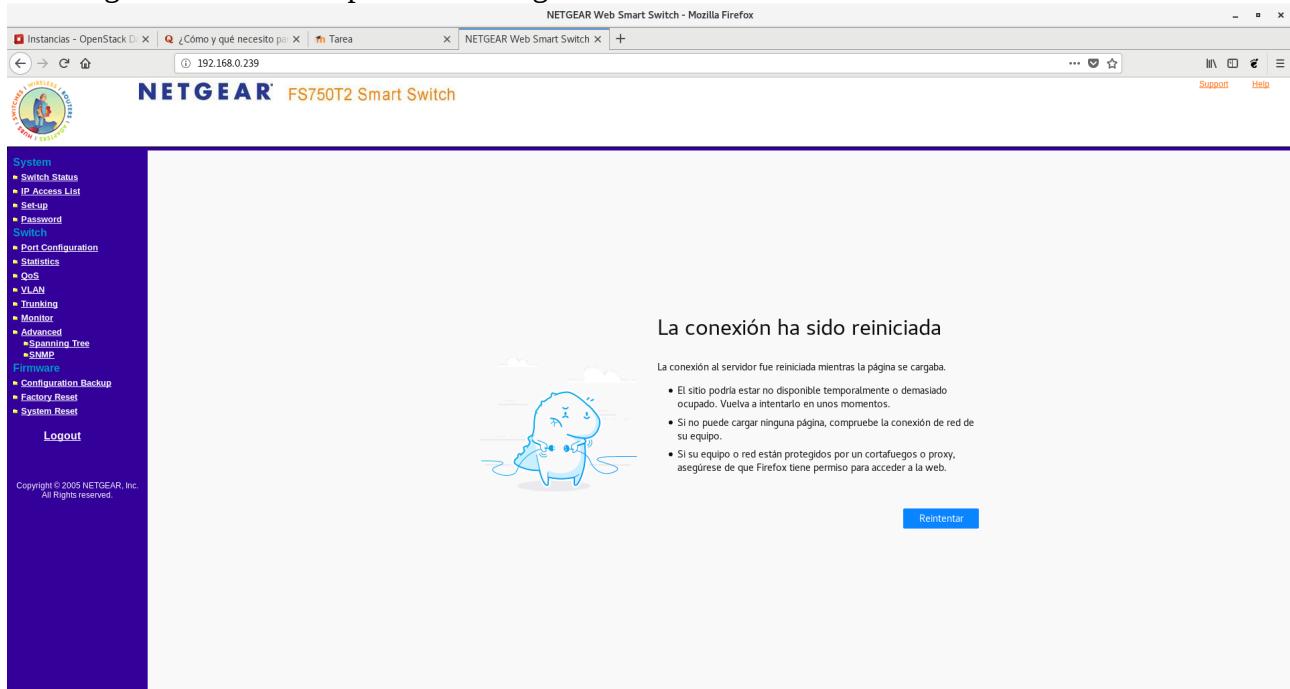
Para cambiar de IP debemos usar la opción de Set-Up



Y insertar la IP estática con máscara /24



Acto seguido el Switch se pondrá a configurar la IP.



## 4 Cambio de IP Maquinas

Para hacernos ping entre nosotros debemos ponernos todos en el mismo rango de IP, en este caso 10.90.10.x/24.

## 5 Captura de Tráfico

Para realizar el Port Mirroring deberemos ir a la configuración de nuestro Switch(Monitor) y elegir el correspondiente puerto que queramos visualizar.

The screenshot shows the NETGEAR Web Smart Switch interface in Microsoft Internet Explorer. The URL in the address bar is <http://10.90.10.106/>. The main title is "NETGEAR® FS750T2 Smart Switch". On the left, there's a sidebar with navigation links including "IP Access List", "Set-up", "Password", "Switch", "Port Configuration", "Statistics", "QoS", "VLAN", "Trunking", "Monitor", "Advanced", "Firmware", "Configuration Backup", and "Factory Reset". The main content area has two dropdown menus: "Sniffer Mode" set to "Both" and "Sniffer Port" set to "47". Below these are two tables of ports. The first table, "Source Port", lists ports 01 through 50. The second table, "Destination Port", lists ports 01 through 12. In the "Destination Port" table, ports 46 and 47 are checked, indicating they are selected for mirroring. At the bottom right are "Apply" and "Help" buttons.

Para ver el tráfico, el que realiza el port Mirroring, Alejandro en este caso, deberá capturarla, ya que el usuario Miguel va a realizar un ping hacia el usuario Luis

Miguel Port 48 (10.90.10.104)

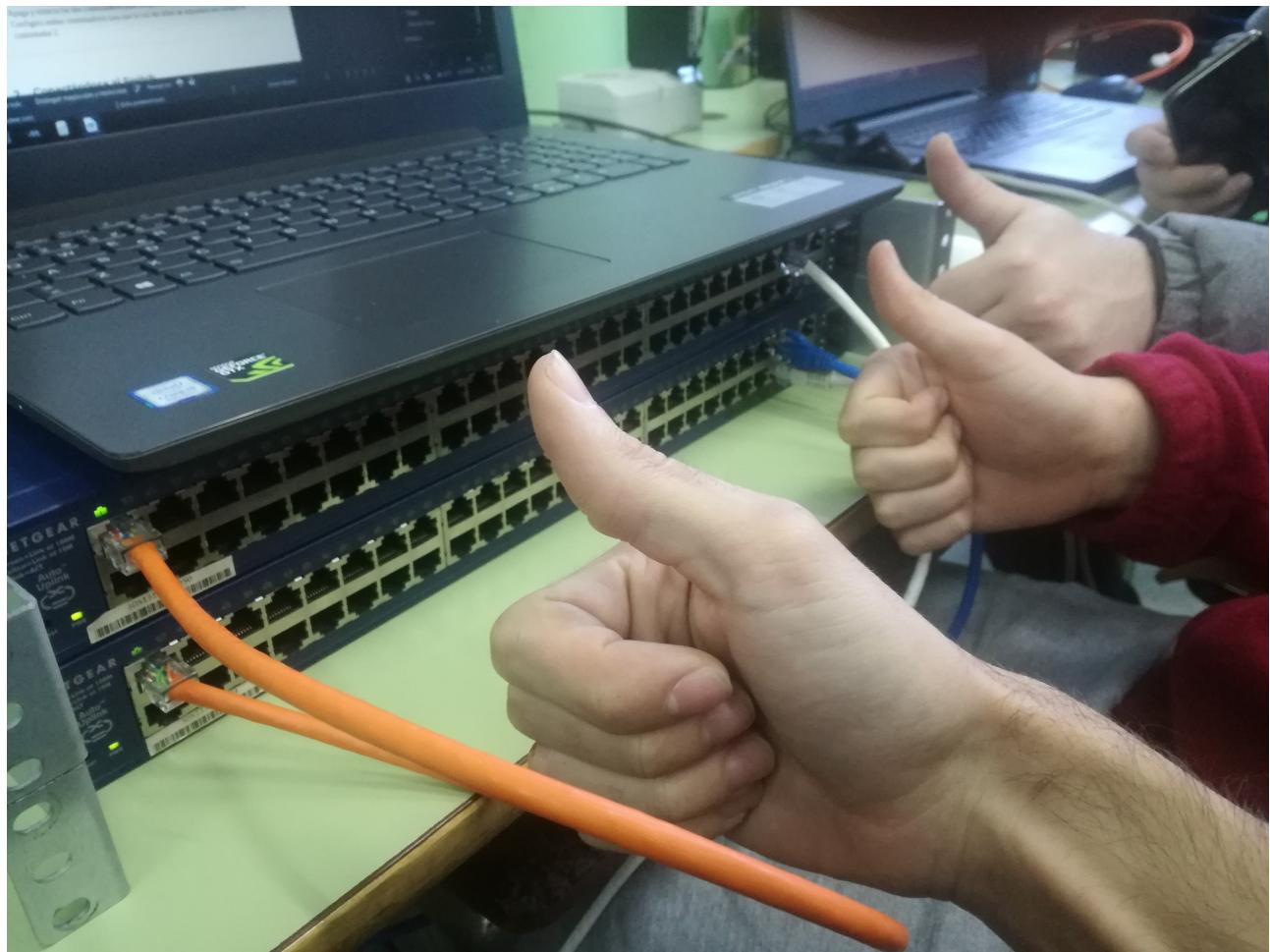
Alejandro Port 47(10.90.10.100)

Luis Port 46(10.90.10.6)

```
alexrr@pc-alex:~$ sudo tcpdump -nni enp7s0 icmp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp7s0, link-type EN10MB (Ethernet), capture size 262144 bytes
^C
0 packets captured
0 packets received by filter
0 packets dropped by kernel
alexrr@pc-alex:~$ sudo tcpdump -nni enp7s0 icmp -c 10
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp7s0, link-type EN10MB (Ethernet), capture size 262144 bytes
09:26:22.726382 IP 10.90.10.6 > 10.90.10.104: ICMP echo request, id 24230, seq 1, length 64
09:26:22.726391 IP 10.90.10.6 > 10.90.10.104: ICMP echo request, id 24230, seq 1, length 64
09:26:22.726528 IP 10.90.10.104 > 10.90.10.6: ICMP echo reply, id 24230, seq 1, length 64
09:26:22.726531 IP 10.90.10.104 > 10.90.10.6: ICMP echo reply, id 24230, seq 1, length 64
09:26:23.726751 IP 10.90.10.6 > 10.90.10.104: ICMP echo request, id 24230, seq 2, length 64
09:26:23.726759 IP 10.90.10.6 > 10.90.10.104: ICMP echo request, id 24230, seq 2, length 64
09:26:23.727271 IP 10.90.10.104 > 10.90.10.6: ICMP echo reply, id 24230, seq 2, length 64
09:26:23.727284 IP 10.90.10.104 > 10.90.10.6: ICMP echo reply, id 24230, seq 2, length 64
09:26:24.741754 IP 10.90.10.6 > 10.90.10.104: ICMP echo request, id 24230, seq 3, length 64
09:26:24.741766 IP 10.90.10.6 > 10.90.10.104: ICMP echo request, id 24230, seq 3, length 64
10 packets captured
10 packets received by filter
0 packets dropped by kernel
alexrr@pc-alex:~$
```

## 6 STP

Conectamos los latiguillos a el puerto 1 y los conectamos así:



Conectamos a dos usuarios al puerto 10 de cada conmutador y comprobamos el ping:

```
root@Miguel:/home/miguelcordoba# tcpdump -nni eth0 icmp -c 10
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link type EN10MB (Ethernet), capture size 262144 bytes
09:49:45.820811 IP 10.90.10.6 > 10.90.10.104: ICMP echo request, id 14927, seq 3, length 64
09:49:45.820891 IP 10.90.10.104 > 10.90.10.6: ICMP echo reply, id 14927, seq 3, length 64
09:49:46.848936 IP 10.90.10.6 > 10.90.10.104: ICMP echo request, id 14927, seq 4, length 64
09:49:46.849916 IP 10.90.10.104 > 10.90.10.6: ICMP echo reply, id 14927, seq 4, length 64
09:49:47.872703 IP 10.90.10.6 > 10.90.10.104: ICMP echo request, id 14927, seq 5, length 64
09:49:47.872778 IP 10.90.10.104 > 10.90.10.6: ICMP echo reply, id 14927, seq 5, length 64
09:49:48.873169 IP 10.90.10.6 > 10.90.10.104: ICMP echo request, id 14927, seq 6, length 64
09:49:48.873236 IP 10.90.10.104 > 10.90.10.6: ICMP echo reply, id 14927, seq 6, length 64
09:49:49.887993 IP 10.90.10.6 > 10.90.10.104: ICMP echo request, id 14927, seq 7, length 64
09:49:49.888060 IP 10.90.10.104 > 10.90.10.6: ICMP echo reply, id 14927, seq 7, length 64
10 packets captured
10 packets received by filter
0 packets dropped by kernel
root@Miguel:/home/miguelcordoba#
```

Y hacemos que ambos usuarios hagan ping al broadcast contrario:

```
luis@luis:~/Escritorio$ ping 10.90.10.106 -c 5
PING 10.90.10.106 (10.90.10.106) 56(84) bytes of data.
64 bytes from 10.90.10.106: icmp_seq=1 ttl=255 time=6.47 ms
64 bytes from 10.90.10.106: icmp_seq=2 ttl=255 time=6.53 ms
64 bytes from 10.90.10.106: icmp_seq=3 ttl=255 time=6.51 ms
64 bytes from 10.90.10.106: icmp_seq=4 ttl=255 time=6.44 ms
64 bytes from 10.90.10.106: icmp_seq=5 ttl=255 time=6.25 ms

--- 10.90.10.106 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 6.259/6.445/6.534/0.131 ms
luis@luis:~/Escritorio$
```

```
root@Miguel:/home/miguelcordoba# ping 10.90.10.112 -c 10
PING 10.90.10.112 (10.90.10.112) 56(84) bytes of data.
64 bytes from 10.90.10.112: icmp_seq=1 ttl=255 time=6.38 ms
64 bytes from 10.90.10.112: icmp_seq=2 ttl=255 time=6.72 ms
64 bytes from 10.90.10.112: icmp_seq=3 ttl=255 time=6.31 ms
64 bytes from 10.90.10.112: icmp_seq=4 ttl=255 time=6.70 ms
64 bytes from 10.90.10.112: icmp_seq=5 ttl=255 time=6.54 ms
64 bytes from 10.90.10.112: icmp_seq=6 ttl=255 time=6.49 ms
64 bytes from 10.90.10.112: icmp_seq=7 ttl=255 time=6.87 ms
64 bytes from 10.90.10.112: icmp_seq=8 ttl=255 time=6.74 ms
64 bytes from 10.90.10.112: icmp_seq=9 ttl=255 time=6.88 ms
64 bytes from 10.90.10.112: icmp_seq=10 ttl=255 time=6.43 ms

--- 10.90.10.112 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9017ms
rtt min/avg/max/mdev = 6.316/6.610/6.884/0.202 ms
root@Miguel:/home/miguelcordoba#
```

Para ver la raíz de los conmutadores deberemos configurar el Spanning Tree y quien tenga el número más bajo será la raíz.

The screenshot shows the NETGEAR Web Smart Switch interface in Microsoft Internet Explorer. The URL in the address bar is <http://10.90.10.106/>. The main content area displays the NETGEAR logo and "FS750T2 Smart Switch". On the left, a sidebar menu under "System" includes "Switch Status", "IP Access List", "Set-up", "Password", "Switch", "Port Configuration", and "Statistics". The right side shows a table with the following configuration settings:

Fast Link	<input type="radio"/> Disable <input checked="" type="radio"/> Enable
Bridge Priority(0 - 65535)	16000 (Default:32768)
Bridge Max Age(6 - 40)	20 (Default:20)
Bridge Hello Time(1 - 10)	2 (Default:2)
Bridge Forward Delay(4 - 30)	15 (Default:15)

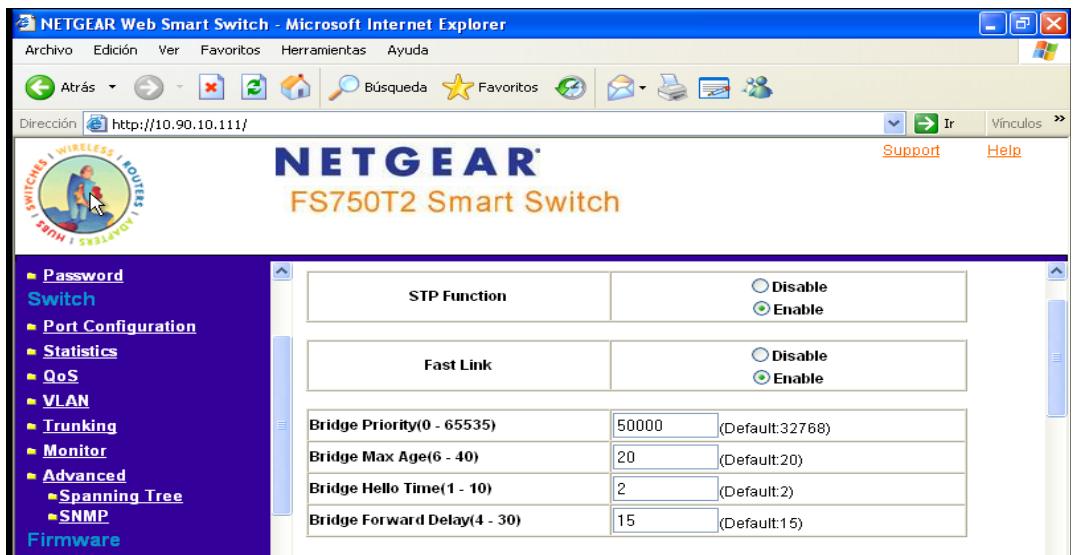
Reiniciamos y comprobamos que efectivamente, se mantiene la misma configuración.

Para asignar una raíz simplemente debemos asignarle a un conmutador mas prioridad que el otro.

En este caso el conmutador 2 tiene prioridad 16000, mientras el otro tendrá prioridad 50000, es decir, el conmutador 2 será la raíz.

The screenshot shows the same NETGEAR Web Smart Switch interface after changes have been made. The configuration table now shows:

Fast Link	<input type="radio"/> Disable <input checked="" type="radio"/> Enable
Bridge Priority(0 - 65535)	50000 (Default:32768)
Bridge Max Age(6 - 40)	20 (Default:20)
Bridge Hello Time(1 - 10)	2 (Default:2)
Bridge Forward Delay(4 - 30)	15 (Default:15)



## 7 Conclusión

Hemos aprendido a configurar el Spanning Tree de un Switch junto a captura de tráfico entre varios puertos mediante el Port Mirroring.