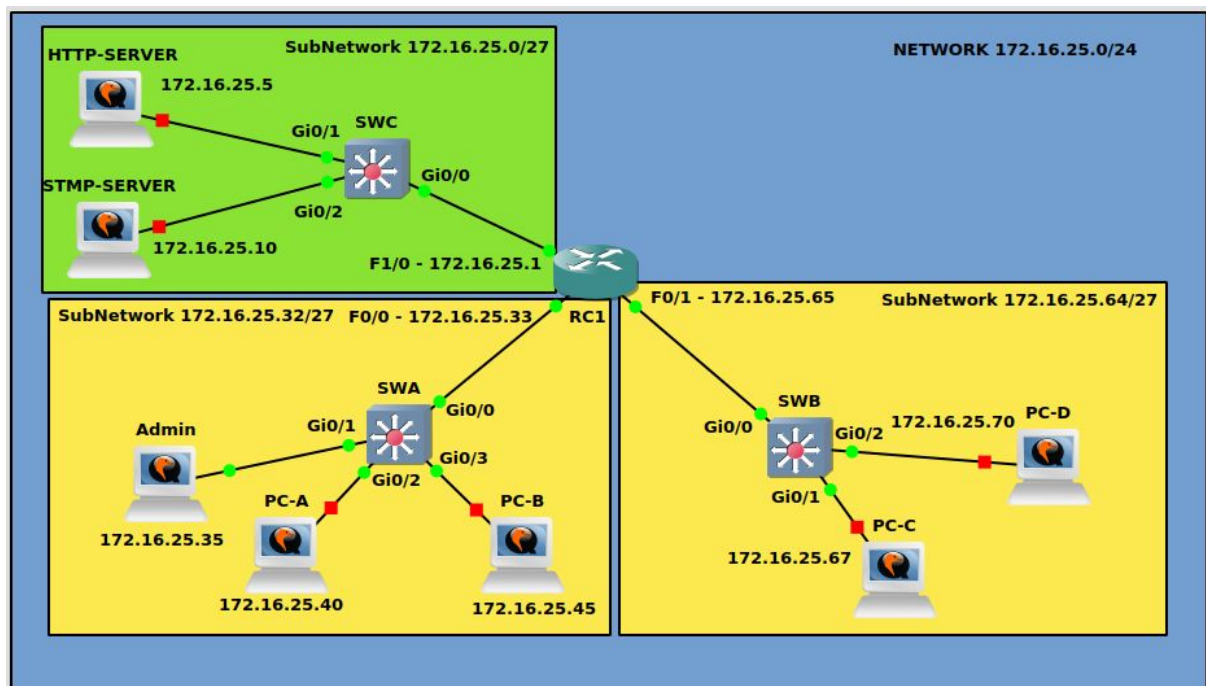


## Protección de la LAN Dispositivos de capa 2

### Consideraciones

- Se ha definido una red 172.16.25.0/24 global para la empresa.
- Se han definido subredes. De momento, la empresa cuenta con 3 subredes, por lo que harían falta 2 bits de direccionamiento adicional (/26), debido a la previsión de futuro se ha habilitado 1 bit más para tener direccionamiento para futuras subredes (/27). Quedando la topología inicial de red de la siguiente manera.



Sobre esta configuración inicial se van a llevar a cabo las configuraciones de protección de dispositivos de capa 2 vistas en el Tema 4: Protección de la LAN.

La configuración del Router Central RC1 se muestra en las siguientes páginas, como se puede observar, se han tenido en cuenta las configuraciones recomendadas, tanto de seguridad, como de funcionamiento de los temas anteriores.

```
upgrade fpd auto
version 15.2
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname RC1
!
boot-start-marker
boot-end-marker
!
!
security authentication failure rate 30 log
enable secret 9
$9$a8ZneQPhZZjcDh$caQy.wQfV7qnG7ohcJNGvS5kJ0/p00C1BMfLPVCz01U
!
no aaa new-model
no ip icmp rate-limit unreachable
!
!
no ip domain lookup
ip domain name mynetwork.com
ip cef
login block-for 60 attempts 5 within 45
login delay 10
login quiet-mode access-class ALLOW-ADMIN
login on-failure log
login on-success log
no ipv6 cef
!
multilink bundle-name authenticated
!
!
username manuelto privilege 15 secret 9
$9$xyd5NDUYFbtFy4$pGNU7Bt0HHTRXq9s8Ctxlr9wKSdbPpZVUXZBtb0BEDU
!
redundancy
!
!
ip tcp synwait-time 5
ip ssh time-out 30
ip ssh version 2
!
```

```
!  
!  
interface FastEthernet0/0  
  ip address 172.16.25.33 255.255.255.224  
  duplex auto  
  speed auto  
!  
interface FastEthernet0/1  
  no ip address  
  shutdown  
  duplex auto  
  speed auto  
!  
interface FastEthernet1/0  
  ip address 172.16.25.1 255.255.255.224  
  duplex half  
!  
ip forward-protocol nd  
no ip http server  
no ip http secure-server  
!  
!  
ip access-list extended ALLOW-ADMIN  
  remark Allow only Administrative Hosts  
  permit ip host 172.16.25.35 any  
!  
no cdp log mismatch duplex  
!  
!  
control-plane  
!  
!  
mgcp profile default  
!  
!  
gatekeeper  
  shutdown  
!  
banner login ^CEsta prohibido iniciar sesion a usuarios no autorizados.  
Se tomaran medidas legales contra los que no respeten esta norma^C  
!  
line con 0  
  exec-timeout 0 0  
  privilege level 15  
  logging synchronous
```

```

login local
stopbits 1
line aux 0
exec-timeout 0 0
privilege level 15
logging synchronous
login local
stopbits 1
line vty 0 4
access-class ALLOW-ADMIN in
login local
transport input telnet ssh
!
end

```

## 1. Port Security

Port Security nos puede ayudar a proteger nuestra red de ataques dirigidos a desbordar la tabla CAM, capturando el tráfico de otras VLANs y empeorando el rendimiento de la red. También podemos defendernos de ataques de inanición DHCP hasta cierto punto.

### Consideraciones

- He decidido definir como máximo 2 direcciones MAC por puerto.
- Los puertos que van dirigidos a dispositivos finales serán puestos en modo de acceso.
- El caso del puerto Gi0/1 del Switch SWA es especial, ya que el PC que conecta es el del administrador de red, por lo que voy a definir su MAC de forma estática, la que queda restante, se aprenderá de manera automática.

Voy a mostrar únicamente la configuración de los puertos Gi0/1 y Gi0/2, ya que el primero es especial y el segundo es el que representa la configuración de puerto básica de la red.

```

interface GigabitEthernet0/1
switchport mode access
switchport port-security maximum 2
switchport port-security mac-address 0c5c.5724.ce00
switchport port-security mac-address sticky
switchport port-security
switchport port-security aging time 30
switchport port-security aging type inactivity
media-type rj45
negotiation auto
!
interface GigabitEthernet0/2
switchport mode access

```

```

switchport port-security maximum 2
switchport port-security mac-address sticky
switchport port-security mac-address sticky 0c5c.5754.f100
switchport port-security aging time 30
switchport port-security aging type inactivity
switchport port-security
media-type rj45
negotiation auto

```

A continuación se muestra el estado de la seguridad del puerto para el gi0/2 en el Switch SWA.

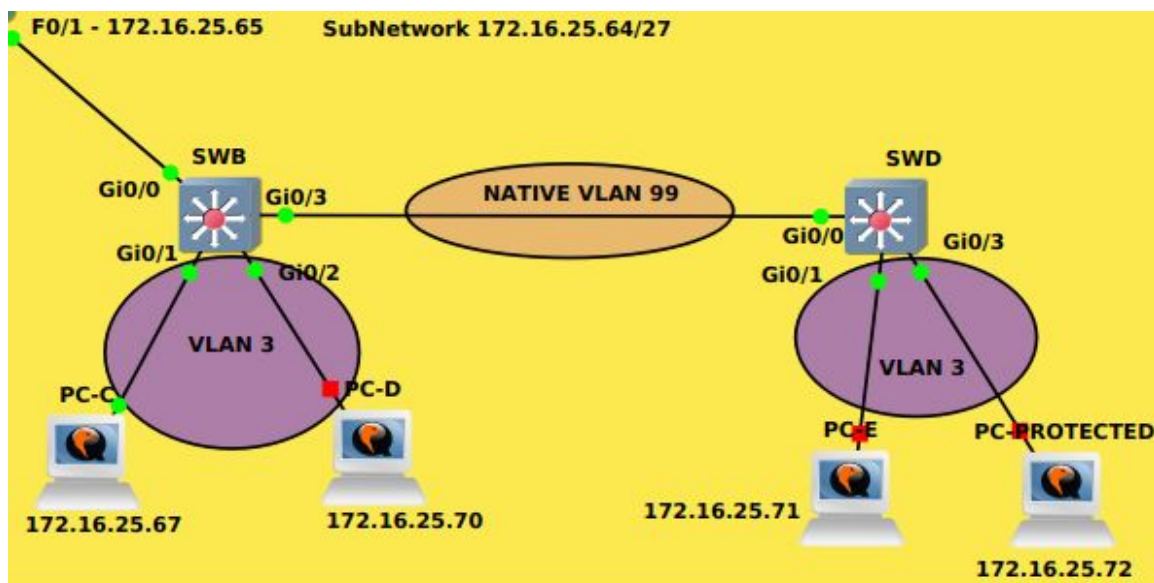
```

Switch#show port-security interface gi0/2
Port Security           : Enabled
Port Status             : Secure-up
Violation Mode          : Shutdown
Aging Time              : 30 mins
Aging Type              : Inactivity
SecureStatic Address Aging : Disabled
Maximum MAC Addresses   : 2
Total MAC Addresses     : 1
Configured MAC Addresses : 0
Sticky MAC Addresses    : 1
Last Source Address:Vlan : 0c5c.5754.f100:1
Security Violation Count : 0

```

## 2. Configuración VLAN y PVLAN

Para este supuesto se van a configurar los switches SWB y SWC como troncales de una red que maneja varias VLANs. A destacar la VLAN 3, que será la funcional entre los puertos asociados a un host de SWB y SWC. Más adelante, esta se dividirá para implementar el enfoque PVLAN.



Se debe resaltar que la configuración de las VLANs no es la específica a seguir en una red real que siga esta topología, ya que tiene carácter de prueba.

Se han realizado las siguientes configuraciones en SWB

```
SWB(config)# interface gi0/3
SWB(config-if)# switchport trunk encapsulation dot1q
SWB(config-if)# switchport mode trunk
SWB(config-if)# switchport nonegotiate
SWB(config-if)# switchport access vlan 99*
SWB(config-if)# switchport vlan native vlan 99
```

**\*Nota:** Es importante definir esta línea, ya que si no se pone y alguna vez el puerto deja de estar en modo trunk pasará a la vlan 1 por defecto.

```
Switch#show interfaces gi0/3 switchport
Name: Gi0/3
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: trunk
Administrative Trunking Encapsulation: isl
Operational Trunking Encapsulation: isl
Negotiation of Trunking: Off
Access Mode VLAN: 99 (VLAN0099)
Trunking Native Mode VLAN: 99 (VLAN0099)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: 99
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
```

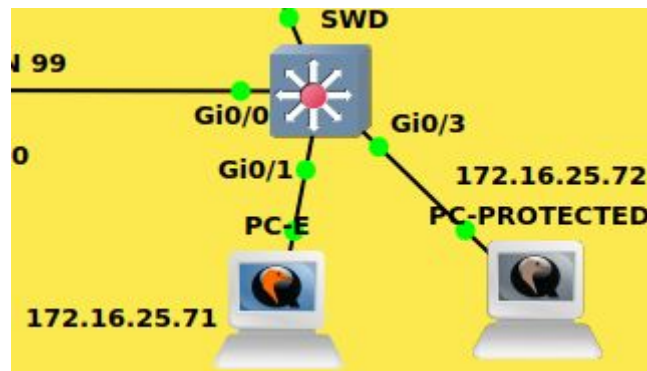
El resto de puertos han sido asociados a una vlan sin uso (34) y se han deshabilitado, como se puede ver en el ejemplo de la siguiente captura.

```
interface GigabitEthernet1/1
 switchport access vlan 34
 switchport mode access
 shutdown
 media-type rj45
 negotiation auto
!
interface GigabitEthernet1/2
 switchport access vlan 34
 switchport mode access
 shutdown
 media-type rj45
 negotiation auto
!
```



La configuración que se ha seguido en el SWD ha sido similar.

## 2.1 Puertos protegidos.



Se ha configurado el puerto Gi0/3 como protegido para probar la funcionalidad. De esta forma no podrá enviar tráfico a través del switch a PC-E, solamente podría a través del Router Central RC1. La configuración se ha realizado de la siguiente forma:

```
SWD(config)# int gi0/3
SWD(config-if)# switchport mode access
SWD(config-if)# switchport access vlan 10
SWD(config-if)# switchport protected
```

Podemos comprobarlo a través del siguiente comando.

```
Switch#show int gi0/3 switchport | include Protected
Protected: true
```


Si ponemos el puerto Gi0/1 como protected también, PC-E y PC-Protected podrán seguir comunicándose, ya que el router reenviará el tráfico entre ellos, pero NO a nivel de enlace. Podemos comprobarlo haciendo un shutdown en el puerto Gi0/0.

Existe otra opción para evitar que se utilice el Router como un proxy. Esta consistiría en declarar una ACL en la que se deniegue el tráfico entre PC-PROTECTED y PC-E (Por ejemplo). La configuración en el router RC1 para la ACL ha sido la siguiente:

```

interface FastEthernet0/1
 ip address 172.16.25.65 255.255.255.224
 ip access-group 199 in
 duplex auto
 speed auto
!
interface FastEthernet1/0
 ip address 172.16.25.1 255.255.255.224
 duplex half
!
ip forward-protocol nd
no ip http server
no ip http secure-server
!
!
ip access-list extended ALLOW-ADMIN
 remark Allow only Administrative Hosts 172.16.25.71
 permit ip host 172.16.25.35 any
!
access-list 199 remark Deny traffic between PC-E and PC-PROTECTED
access-list 199 deny ip host 172.16.25.72 host 172.16.25.71
no cdp log mismatch duplex

```



Obteniendo, ahora sí, un resultado negativo cuando se intenta hacer ping desde el PC-Protected al PC-E.

```

[osboxes@osboxes ~]$ ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.16.25.72 netmask 255.255.255.224 broadcast 172.16.25.95
    inet6 fe80::d3db:6e7c:1d32:ea73 prefixlen 64 scopeid 0x20<link>
    ether 0c:5c:57:3b:8c:00 txqueuelen 1000 (Ethernet)
    RX packets 49 bytes 3782 (3.6 KiB)
    RX errors 0 dropped 5 overruns 0 frame 0
    TX packets 408 bytes 33487 (32.7 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[osboxes@osboxes ~]$ ping 172.16.25.71
PING 172.16.25.71 (172.16.25.71) 56(84) bytes of data.
From 172.16.25.72 icmp_seq=1 Destination Host Unreachable
From 172.16.25.72 icmp_seq=2 Destination Host Unreachable
From 172.16.25.72 icmp_seq=3 Destination Host Unreachable
From 172.16.25.72 icmp_seq=4 Destination Host Unreachable
^C
--- 172.16.25.71 ping statistics ---
4 packets transmitted, 0 received, +4 errors, 100% packet loss, time 3000ms
pipe 4
[osboxes@osboxes ~]$ ping -c 2 172.16.25.67
PING 172.16.25.67 (172.16.25.67) 56(84) bytes of data.
64 bytes from 172.16.25.67: icmp_seq=1 ttl=64 time=9.63 ms
64 bytes from 172.16.25.67: icmp_seq=2 ttl=64 time=10.2 ms

--- 172.16.25.67 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 9.637/9.959/10.281/0.322 ms
[osboxes@osboxes ~]$

```



En la captura vemos como no podemos hacer ping al PC-E, pero sí a PC-C, ya que es miembro de nuestra VLAN.

## 2.2 Configuración PVLAN.

En este apartado configuraremos la VLAN 3 del switch SWB como privada, definiendo los siguientes apartados:

1. VLAN 3 será la primaria.
2. VLAN 501 será secundaria en modo isolated.
3. VLAN 502 será secundaria en modo community.
4. VLAN 503 será secundaria en modo community.

Las comandos a realizar son los siguientes:

```
SWB(config)# vtp mode transparent
SWB(config)# vlan 3
SWB(config-vlan)# private-vlan primary
SWB(config-vlan)# exit
SWB(config)# vlan 501
SWB(config-vlan)# private-vlan isolated
SWB(config-vlan)# exit
SWB(config-vlan)# vlan 502
SWB(config-vlan)# private-vlan community
SWB(config-vlan)# exit
SWB(config)# vlan 3
SWB(config-vlan)# private-vlan association 501-503
```

Ahora que están definidas las PVLAN, solo falta asociarlas al puerto que queramos. Para ello realizamos los siguientes comandos:

```
SWB(config)# int gi0/2
SWB(config-if)# switchport mode private-vlan host
SWB(config-if)# switchport private-vlan host-association 3 502
SWB(config-if)# exit
SWB(config)# int gi0/1
SWB(config-if)# switchport mode private-vlan host
SWB(config-if)# switchport private-vlan host-association 3 502
SWB(config-if)# exit
SWB(config)# int gi0/0
SWB(config-if)# switchport mode private-vlan promiscuous
SWB(config-if)# switchport private-vlan mapping 3 add 501-503
SWB(config-if)# end
SWB(config)# interface vlan 3
SWB(config-if)# ip address 172.16.25.66 255.255.255.224
```

```
SWB(config-if)# no shutdown
```

De esta forma el Router Central RC1 está en modo promiscuo y PC-C-D están en modo community, por lo que entre los tres se pueden comunicar. Ahora bien, si cambiamos el estado de PC-D a isolated veremos que PC-D solo se podrá comunicar con el Router. Para ello aplicamos los siguientes comandos.

```
SWB(config)# int gi0/2
SWB(config-if)# no switchport private-vlan host-association 3 502
SWB(config-if)# no switchport private-vlan host-association 3 501
SWB(config-if)# exit
```

Como vemos en la siguiente captura, PC-D puede enviar paquetes al Router RC1, pero no a RCE.

```
[osboxes@osboxes ~]$ ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.16.25.70 netmask 255.255.255.224 broadcast 172.16.25.95
    inet6 fe80::15d5:5289:777b:8c7f prefixlen 64 scopeid 0x20<link>
    ether 0c:5c:57:f5:d2:00 txqueuelen 1000 (Ethernet)
    RX packets 82 bytes 5918 (5.7 KiB)
    RX errors 0 dropped 5 overruns 0 frame 0
    TX packets 135 bytes 11979 (11.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[osboxes@osboxes ~]$ ping -c 2 172.16.25.67
PING 172.16.25.67 (172.16.25.67) 56(84) bytes of data.
From 172.16.25.70 icmp_seq=1 Destination Host Unreachable
From 172.16.25.70 icmp_seq=2 Destination Host Unreachable

--- 172.16.25.67 ping statistics ---
2 packets transmitted, 0 received, +2 errors, 100% packet loss, time 1001ms
pipe 2
[osboxes@osboxes ~]$ ping -c 2 172.16.25.65
PING 172.16.25.65 (172.16.25.65) 56(84) bytes of data.
64 bytes from 172.16.25.65: icmp_seq=1 ttl=255 time=11.9 ms
64 bytes from 172.16.25.65: icmp_seq=2 ttl=255 time=10.2 ms

--- 172.16.25.65 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 10.205/11.059/11.913/0.854 ms
[osboxes@osboxes ~]$
```

Podemos comprobar la configuración final en el Switch SWB con el siguiente comando:

```
RC1# show vlan
```

Primary	Secondary	Type	Ports
3	501	isolated	Gi0/0, Gi0/2
3	502	community	Gi0/0, Gi0/1
3	503	community	Gi0/0

### 3. DHCP SNOOPING

DHCP Snooping es un mecanismo que nos ayudará a asociar puertos confiables y no confiables en cuanto a la recepción de paquetes DHCP. De esta manera podremos evitar suplantaciones DHCP e inaniciones DHCP.

Para este ejemplo, se va a entender que un puerto es confiable si puede recibir paquetes DHCP por parte del Router RC1, el único dispositivo que puede asignar IP's a otros dispositivos en esta red. La configuración del servidor DHCP se detalla a continuación:

```
RC1(config)# ip dhcp pool 1
RC1(dhcp-config)#utilization mark high 80 log
RC1(dhcp-config)#network 172.16.25.64 /27
RC1(dhcp-config)#domain-name mynetwork.com
RC1(dhcp-config)#default-router 172.16.25.65
RC1(dhcp-config)#lease 30
RC1(dhcp-config)#end
RC1(config)#ip dhcp pool 1
RC1(dhcp-config)#host 172.16.25.70
RC1(dhcp-config)#client-identifier clientD
% DHCP could not parse the client identifier. Check character 0 (c).
RC1(dhcp-config)#client-identifier 01b7.0813.8811.66
RC1(dhcp-config)#hardware-address 0c5c.57f5.d200 ethernet
RC1(dhcp-config)#client-name osboxes
RC1(dhcp-config)#end
```

Si pruebo a conectarme a la red mediante DHCP desde el cliente D, veo que efectivamente me proporciona la IP que he definido. Podemos ver las estadísticas del servidor DHCP en el router RC1 con el siguiente comando:

```

RC1#show ip dhcp server statistics
Memory usage      25041
Address pools     2
Database agents   0
Automatic bindings 0
Manual bindings   1
Expired bindings  0
Malformed messages 0
Secure arp entries 0

Message           Received
BOOTREQUEST       0
DHCPDISCOVER      1
DHCPRREQUEST      1
DHCPDECLINE       0
DHCPRELEASE       0
DHCPINFORM        0

Message           Sent
BOOTREPLY         0
DHCPOFFER         1
DHCPACK           1
DHCPNAK           0
RC1#

```

Se ha configurado DHCP Snooping en el switch SWB a través de los siguientes comandos:

```

Switch(config-if)#ip dhcp snooping trust
Switch(config)#int range gi0/1-3
Switch(config-if-range)#ip dhcp snooping limit rate 6
Switch(config-if-range)#exit
Switch(config)#int range gi1/0-3
Switch(config-if-range)#ip dhcp snooping limit rate 6
Switch(config-if-range)#exit
Switch(config)#int range gi2/0-3
Switch(config-if-range)#ip dhcp snooping limit rate 6
Switch(config-if-range)#exit
Switch(config)#int range gi3/0-3
Switch(config-if-range)#ip dhcp snooping limit rate 6
Switch(config-if-range)# ip dhcp snooping vlan 3

```

Podemos comprobar que el comando se ha ejecutado con éxito a través del siguiente comando:

```
Switch#show ip dhcp snooping
Switch DHCP snooping is enabled
Switch DHCP gleaning is disabled
DHCP snooping is configured on following VLANs:
3
DHCP snooping is operational on following VLANs:
3,501-503
DHCP snooping is configured on the following L3 Interfaces:

Insertion of option 82 is enabled
  circuit-id default format: vlan-mod-port
  remote-id: 0c5c.573b.5100 (MAC)
Option 82 on untrusted port is not allowed
Verification of hwaddr field is enabled
Verification of giaddr field is enabled
DHCP snooping trust/rate is configured on the following Interfaces:
```

Interface	Trusted	Allow option	Rate limit (pps)
GigabitEthernet0/0	no	no	6
GigabitEthernet0/1	no	no	6
GigabitEthernet0/2	no	no	6
GigabitEthernet0/3	no	no	6
GigabitEthernet1/0	no	no	6
GigabitEthernet1/1	no	no	6
GigabitEthernet1/2	no	no	6
GigabitEthernet1/3	no	no	6
GigabitEthernet2/0	no	no	6
GigabitEthernet2/1	no	no	6

De hecho, si marcamos el puerto Gi0/0 como untrusted para probar esta funcionalidad vemos como no podemos asignar una ip. De hecho, podemos comprobar los paquetes que han sido desechados con el siguiente comando.

```
Switch#show ip dhcp snooping statistics
Packets Forwarded          = 0
Packets Dropped            = 5
Packets Dropped From untrusted ports = 5
Switch#
```

Después de esto se debe de cambiar el puerto Gi0/0 a confiable.



#### 4. DAI (Dynamic ARP Inspection)

La inspección dinámica de mensajes ARP nos ayudará a mitigar ataques de suplantación ARP, ya sea a través de envenenamiento ARP o mecanismos MITM más complejos. Se ha configurado DAI en el Switch SWB a través de los siguientes comandos.

```
Switch(config)#ip arp inspection vlan 3
Switch(config)#ip arp inspection validate src-mac dst-mac ip
Switch(config)#int gi0/0
Switch(config-if)#ip arp inspection trust
```

Tras realizar una serie de peticiones ARP indebidas desde PC-E, podemos comprobar que la configuración funciona de la manera esperada a través del siguiente comando:

```
Switch#show ip arp inspection statistics
```

Vlan	Forwarded	Dropped	DHCP Drops	ACL Drops
3	0	28	28	0
501	0	0	0	0
502	0	0	0	0
503	0	0	0	0

#### 5. IPSG (IP Source Guard)

A través de IPSG podemos minimizar el riesgo de ataques de suplantación de direcciones, sobre todo las dirigidas a la capa 3 del modelo OSI. La configuración que se ha seguido en el Switch SWB es la siguiente.

```
Switch(config)#int range gi1/0-3
Switch(config-if-range)#ip verify source
Switch(config-if-range)#exit
Switch(config)#int range gi2/0-3
Switch(config-if-range)#ip verify source
Switch(config-if-range)#exit
Switch(config)#int ran
Switch(config)#int range gi3/0-
Switch(config)#int range gi3/0-3
Switch(config-if-range)#ip verify source
Switch(config-if-range)#ip verify source
```

```
Switch#show ip verify source
*Dec 6 07:22:19.575: %SYS-5-CONFIG_I: Configured from console by console
Interface  Filter-type  Filter-mode  IP-address      Mac-address      Vlan
-----
Gi0/0      ip            inactive-trust-port
Gi0/1      ip            active        172.16.25.35    1
Gi0/2      ip            active        172.16.25.40    1
Switch#
```

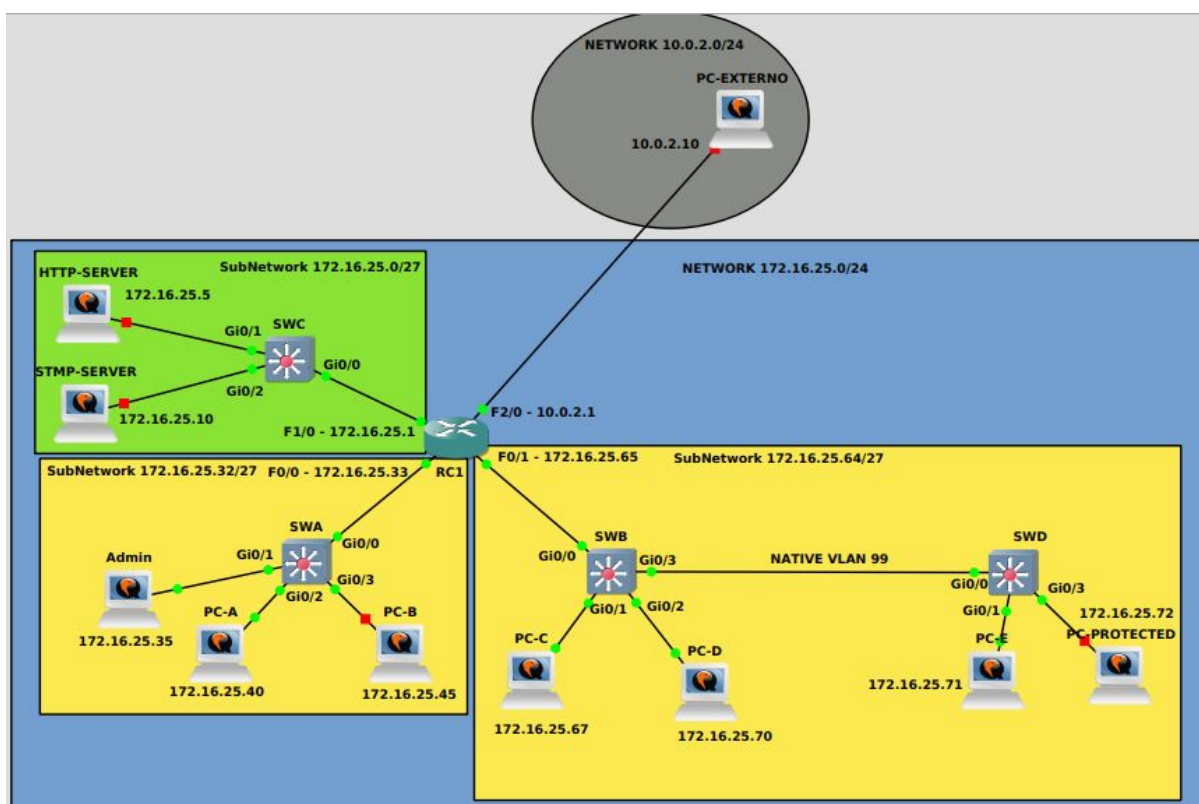
Hay que destacar que los puertos que están marcados como confiables o 'trust' no pueden ser analizados mediante IPSG. Por ello, la configuración correcta sería la siguiente.

```
Switch#show ip verify source
Interface  Filter-type  Filter-mode  IP-address      Mac-address      Vlan
-----
Gi0/1      ip-mac       active        172.16.25.35    0C:5C:57:24:CE:00 1
Gi0/2      ip-mac       active        172.16.25.40    0C:5C:57:54:F1:00 1
Switch#
```

## 5. FIREWALL

### 5.1 Firewall Classic. Enfoque DMZ

En este supuesto se va a configurar un Firewall Classic de Cisco a través del entorno mostrado anteriormente, habiendo añadido una nueva red que hará de red externa o equivalente a Internet. Por otro lado, la subred 172.16.25.0/27 será la red DMZ, mientras que la subred 172.16.25.32/27 será la red privada para este ejemplo.



En primer lugar, se va a llevar a cabo la configuración general de una DMZ:

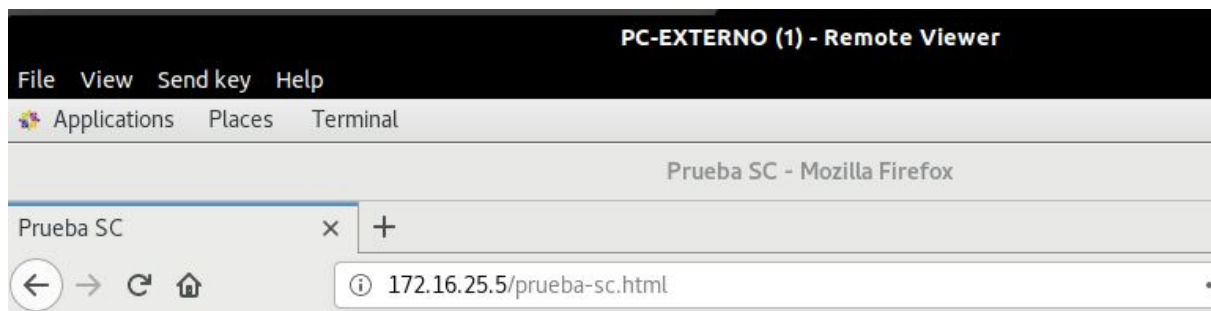
- + Permitiremos el tráfico entrante de la red 10.0.2.0/24 con dirección DMZ y al revés.
- Denegaremos el tráfico originado en la DMZ con dirección red privada.
- + Permitiremos el tráfico desde la red privada a la DMZ.
- + Permitiremos el tráfico desde la red privada a la red pública.
- Denegaremos el tráfico que se origine en la red pública con dirección red privada.

Todo esto aprovechando la funcionalidad de **inspección** de los firewall classic.

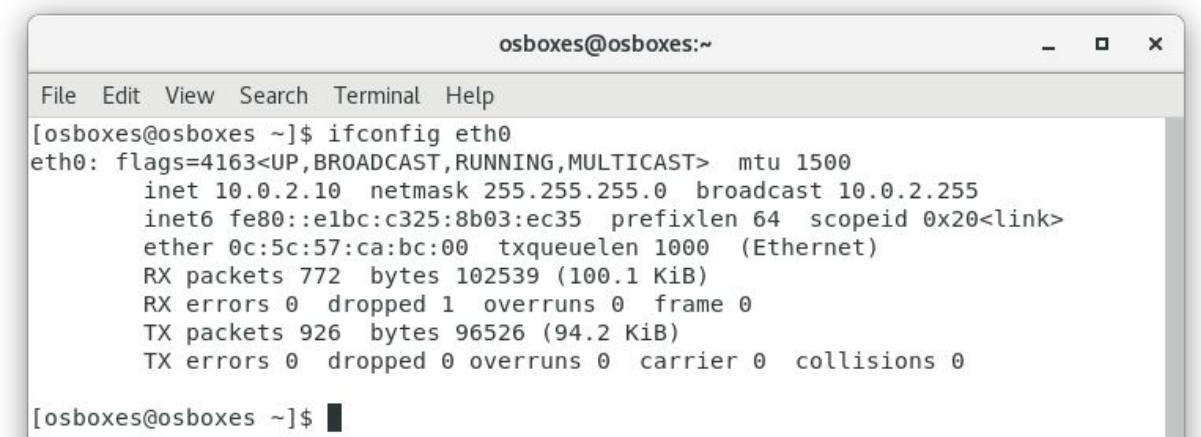
Para llevar a cabo estas configuraciones se han realizado los siguientes comandos en el Router RC1:

```
RC1(config)# ip access-list extended outside-allow
RC1(config-ext-nacl)#permit tcp any 172.16.25.5 0.0.0.0 eq 80
RC1(config-ext-nacl)#permit tcp any 172.16.25.5 0.0.0.0 eq 443
RC1(config-ext-nacl)#permit sm any 172.16.25.10 0.0.0.0 eq 25

RC1(config)#ip inspect name dns-rule dns
RC1(config)#int f1/0
RC1(config-if)#ip ins
RC1(config-if)#ip inspect dns-rule in
```



## Prueba Seguridad de Comunicaciones



En la anterior captura vemos como, tras asignar la ACL seguimos pudiendo acceder desde el PC-EXTERNO al servidor web.

```
RC1(config)#ip access-list extended dmz-allow
RC1(config-ext-nacl)#deny ip 172.16.25.0 0.0.0.31 172.16.25.32 0.0.0.31
RC1(config-ext-nacl)#deny tcp 172.16.25.0 0.0.0.31 172.16.25.1 0.0.0.0
eq 22
RC1(config-ext-nacl)#deny tcp 172.16.25.0 0.0.0.31 172.16.25.1 0.0.0.0
eq 23
RC1(config-ext-nacl)#deny tcp 172.16.25.0 0.0.0.31 172.16.25.1 0.0.0.0
eq 21
RC1(config-ext-nacl)#exit
RC1(config)#int f1/0
RC1(config-if)#ip access-group dmz-allow in

RC1(config)#ip access-list extended private-allow
RC1(config-ext-nacl)#permit tcp 172.16.25.35 0.0.0.0 172.16.25.0
0.0.0.31 eq 22
RC1(config-ext-nacl)#permit tcp 172.16.25.35 0.0.0.0 172.16.25.0
0.0.0.31 eq 23
RC1(config-ext-nacl)#permit tcp 172.16.25.35 0.0.0.0 172.16.25.0
0.0.0.31 eq 21
RC1(config-ext-nacl)#deny tcp 172.16.25.32 0.0.0.31 172.16.25.0 0.0.0.31
eq 22
RC1(config-ext-nacl)#permit tcp 172.16.25.32 0.0.0.31 172.16.25.0
0.0.0.31 eq 80
RC1(config-ext-nacl)#permit tcp 172.16.25.32 0.0.0.31 172.16.25.0
0.0.0.31 eq 443
RC1(config-ext-nacl)#permit tcp 172.16.25.32 0.0.0.31 172.16.25.0
0.0.0.31 eq 25
RC1(config-ext-nacl)#permit icmp 172.16.25.35 0.0.0.0 any echo
RC1(config-ext-nacl)#permit icmp 172.16.25.32 0.0.0.31 any
parameter-problem
RC1(config-ext-nacl)#permit icmp 172.16.25.32 0.0.0.31 any
packet-too-big
RC1(config-ext-nacl)#permit icmp 172.16.25.32 0.0.0.31 any source-quench
RC1(config-ext-nacl)#deny ip 172.16.25.32 0.0.0.31 any

RC1(config)# ip inspect name https-private-dm http
RC1(config)# ip inspect name https-private-dm https
RC1(config)# ip inspect name https-private-dm imap
RC1(config)# ip inspect name https-private-dm ssh
RC1(config)# ip inspect name https-private-dm telnet
RC1(config)# ip inspect name https-private-dm ftp
RC1(config)# ip inspect name https-private-dm icmp echo
RC1(config)# int f0/0
```

```
RC1(config-if)# ip inspect https-private-dm in
```

Es importante destacar que en estas ACL no se tiene en cuenta el tráfico administrativo del router, ya que este es filtrado por la ACL ALLOW-ADMIN asignada a las vty 0 4 que vimos al principio del documento. En las siguientes capturas vemos algunas pruebas del correcto funcionamiento del firewall classic definido.

- DMZ no puede acceder por ella misma a la red privada.

```
[osboxes@osboxes html]$ ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.16.25.5 netmask 255.255.255.224 broadcast 172.16.25.31
    inet6 fe80::6187:a4c3:19e9:bb0b prefixlen 64 scopeid 0x20<link>
    ether 0c:5c:57:b3:b5:00 txqueuelen 1000 (Ethernet)
    RX packets 16940 bytes 1194013 (1.1 MiB)
    RX errors 0 dropped 7 overruns 0 frame 0
    TX packets 17018 bytes 1350184 (1.2 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[osboxes@osboxes html]$ ping -c 2 172.16.25.33
PING 172.16.25.33 (172.16.25.33) 56(84) bytes of data.
From 172.16.25.1 icmp_seq=1 Packet filtered
From 172.16.25.1 icmp_seq=2 Packet filtered
```

- Tanto la red privada (gracias a las inspecciones) como la externa puede acceder a la red DMZ.





- El administrador puede acceder directamente a la red DMZ y red pública para realizar mantenimiento.

```

Admin (1) - Remote Viewer
File View Send key Help
Applications Places Terminal

Prueba SC - Mozilla Firefox
Prueba SC x +
osboxes@osboxes:~
File Edit View Search Terminal Help

PING 172.16.25.5 (172.16.25.5) 56(84) bytes of data.
^C
--- 172.16.25.5 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2000ms

[osboxes@osboxes ~]$ ping 172.16.25.5
PING 172.16.25.5 (172.16.25.5) 56(84) bytes of data.
64 bytes from 172.16.25.5: icmp_seq=1 ttl=63 time=29.4 ms
64 bytes from 172.16.25.5: icmp_seq=2 ttl=63 time=18.4 ms
64 bytes from 172.16.25.5: icmp_seq=3 ttl=63 time=16.4 ms
^C
--- 172.16.25.5 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 16.423/21.445/29.416/5.700 ms
[osboxes@osboxes ~]$ ping 10.0.2.10
PING 10.0.2.10 (10.0.2.10) 56(84) bytes of data.
64 bytes from 10.0.2.10: icmp_seq=1 ttl=63 time=106 ms
64 bytes from 10.0.2.10: icmp_seq=2 ttl=63 time=24.2 ms
64 bytes from 10.0.2.10: icmp_seq=3 ttl=63 time=22.3 ms
^C
--- 10.0.2.10 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 22.349/50.919/106.168/39.074 ms
[osboxes@osboxes ~]$

```

- Acceso administrativo desde DMZ al router está prohibido.

```

[osboxes@osboxes html]$ ssh 172.16.25.1
ssh: connect to host 172.16.25.1 port 22: Connection refused
[osboxes@osboxes html]$ █

```

## 5.2 Firewall ZBF (Zone Based Firewall).

El firewall basado en zonas nos ofrece la posibilidad de realizar una administración y filtrado del tráfico mucho más sencilla y entendible, dividiendo las diferentes partes de la red en zonas, todo ello en función de los requisitos que tengan las mismas.

Para este supuesto se ha clonado la configuración del Router RC1, eliminando las ACL declaradas en el anterior supuesto de las interfaces a usar.

1. El primer paso es definir las zonas que vamos a usar.

```
RC2(config)#zone security PRIVATE
RC2(config-sec-zone)#exit
RC2(config)#zone security PUBLIC
RC2(config-sec-zone)#exit
RC2(config)#zone security DMZ
```

2. Identificar el tráfico para cada par de zonas de la red.

```
RC2(config)#class-map type inspect match-any PRIVATE-OUTSIDE-ALLOWED
RC2(config-cmap)#match protocol http
RC2(config-cmap)#match protocol https
RC2(config-cmap)#match protocol icmp
RC2(config-cmap)#match protocol ssh
RC2(config)#class-map type inspect match-any OUTSIDE-DMZ-ALLOW-PING
RC2(config-cmap)#match protocol icmp
RC2(config-cmap)#match protocol http
RC2(config-cmap)#match protocol https
RC2(config-cmap)#match protocol imap
RC2(config-cmap)#match protocol dns
RC2(config)#class-map type inspect match-any PRIVATE-DMZ-ALLOWED
RC2(config-cmap)#match protocol http
RC2(config-cmap)#match protocol https
RC2(config-cmap)#match protocol icmp
RC2(config-cmap)#match protocol ssh
RC2(config-cmap)#match protocol telnet
RC2(config-cmap)#match protocol ftp
RC2(config-cmap)#match protocol imap
RC2(config-cmap)#match protocol dns
```

3. Definir una acción para este tipo de tráfico.

```
RC2(config-cmap)#policy-map type inspect PRIV-TO-OUT-POLICY
RC2(config-pmap)#class type inspect PRIVATE-OUTSIDE-ALLOWED
RC2(config-pmap-c)#inspect
RC2(config)#policy-map type inspect OUT-TO-DMZ-POLICY
RC2(config-pmap)#class type inspect OUTSIDE-DMZ-ALLOW-PING
RC2(config-pmap-c)#inspect
RC2(config)#policy-map type inspect PRIV-TO-DMZ-POLICY
RC2(config-pmap)#class type
RC2(config-pmap)#class type ins
```

```
RC2(config-pmap)#class type inspect PRIVATE-DMZ-ALLOWED
```

#### 4. Definir pares de zonas o dirección del tráfico.

```
RC2(config)#zone-pair security PRIV-OUT source PRIVATE destination  
PUBLIC  
RC2(config-sec-zone-pair)#service-policy type inspect PRIV-TO-OUT-POLICY  
RC2(config)#zone-pair security OUT-DMZ source PUBLIC destination DMZ  
RC2(config-sec-zone-pair)#service-policy type inspect OUT-TO-DMZ-POLICY  
RC2(config)#zone-pair security PRIV-DMZ source PRIVATE destination DMZ  
RC2(config-sec-zone-pair)#service-policy type inspect PRIV-TO-DMZ-POLICY
```

#### 5. Asignar las zonas a las interfaces.

```
RC2(config)#int f0/0  
RC2(config-if)#zone-member security PRIVATE  
RC2(config-if)#exit  
RC2(config)#int f0/1  
RC2(config-if)#zone security PRIVATE  
RC2(config-if)#exit  
RC2(config)#int f2/0  
RC2(config-if)#zone-member security PUBLIC  
RC2(config-if)#exit  
RC2(config)#int f1/0  
RC2(config-if)#zone-member security DMZ
```

Algo importante a destacar es que el par de zonas Public->Private no necesita la creación de una política específica, ya que al asignarle la zona, por defecto se le asigna una que deniega todo tipo de tráfico.

Se puede comprobar el correcto funcionamiento de la política de zonas definida a través del siguiente comando mostrado en las páginas que vienen a continuación.

```
RC2#show policy-map type inspect zone-pair sessions
```

```
policy exists on zp PRITV-OUT
```

```
Zone-pair: PRITV-OUT
```

```
Service-policy inspect : PRIV-TO-OUT-POLICY
```

```
  Class-map: PRIVATE-OUTSIDE-ALLOWED (match-any)
```

```
    Match: protocol http
```

```
      0 packets, 0 bytes
```

```
      30 second rate 0 bps
```

```
    Match: protocol https
```

```
      0 packets, 0 bytes
```

```
      30 second rate 0 bps
```

```
    Match: protocol icmp
```

```
      2 packets, 128 bytes
```

```
      30 second rate 0 bps
```

```
    Match: protocol ssh
```

```
      0 packets, 0 bytes
```

```
      30 second rate 0 bps
```

```
Inspect
```

```
  Class-map: class-default (match-any)
```

```
    Match: any
```

```
    Drop
```

```
      0 packets, 0 bytes
```

policy exists on zp OUT-DMZ  
Zone-pair: OUT-DMZ

Service-policy inspect : OUT-TO-DMZ-POLICY

Class-map: OUTSIDE-DMZ-ALLOW-PING (match-any)

Match: protocol icmp  
1 packets, 64 bytes  
30 second rate 0 bps

Match: protocol http  
1 packets, 40 bytes  
30 second rate 0 bps

Match: protocol https  
0 packets, 0 bytes  
30 second rate 0 bps

Match: protocol imap  
0 packets, 0 bytes  
30 second rate 0 bps

Match: protocol dns  
0 packets, 0 bytes  
30 second rate 0 bps

Inspect

Class-map: class-default (match-any)

Match: any

Drop

0 packets, 0 bytes



```
policy exists on zp PRIV-DMZ
Zone-pair: PRIV-DMZ

Service-policy inspect : PRIV-TO-DMZ-POLICY

Class-map: PRIVATE-DMZ-ALLOWED (match-any)
  Match: protocol http
    1 packets, 40 bytes
    30 second rate 0 bps
  Match: protocol https
    0 packets, 0 bytes
    30 second rate 0 bps
  Match: protocol icmp
    0 packets, 0 bytes
    30 second rate 0 bps
  Match: protocol ssh
    0 packets, 0 bytes
    30 second rate 0 bps
  Match: protocol telnet
    0 packets, 0 bytes
    30 second rate 0 bps
  Match: protocol ftp
    0 packets, 0 bytes
    30 second rate 0 bps
  Match: protocol imap
    0 packets, 0 bytes
    30 second rate 0 bps
  Match: protocol dns
    0 packets, 0 bytes
    30 second rate 0 bps

Inspect

Class-map: class-default (match-any)
  Match: any
  Drop
    0 packets, 0 bytes
```

## **REFERENCIAS**

<https://www.cisco.com/en/US/docs/>