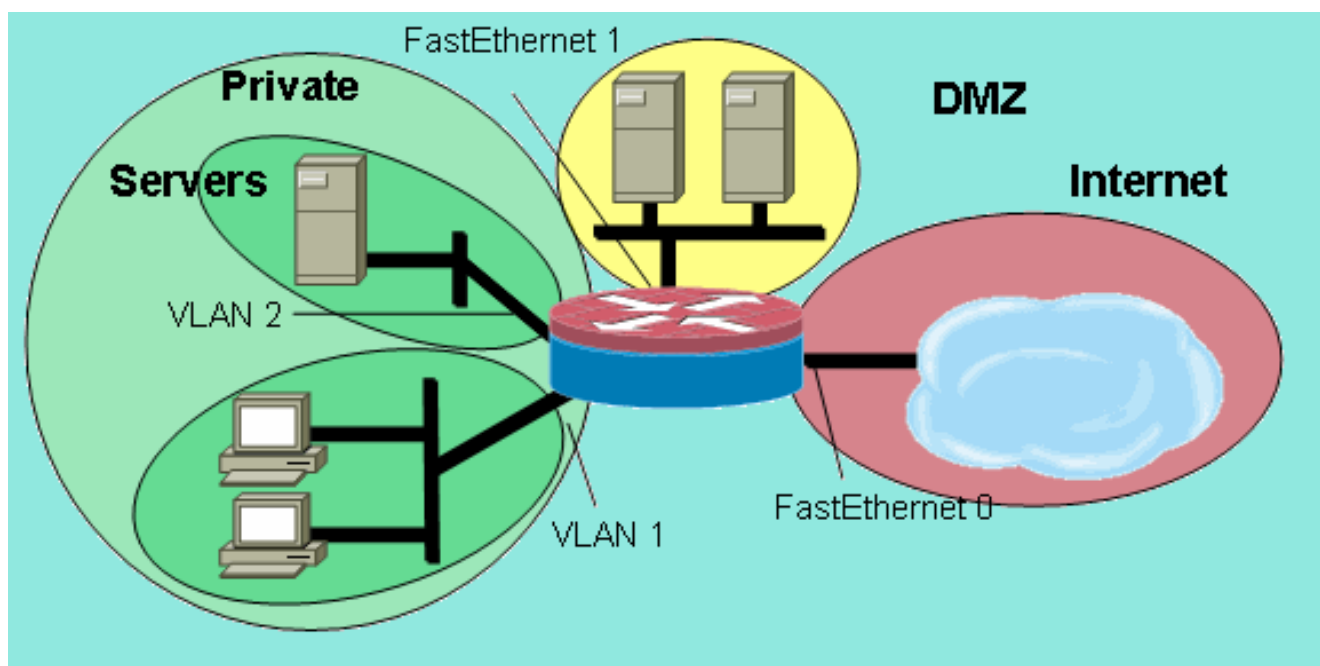


RDE. IES Haría

UT6. Actividad complementaria 2

Práctica

PAT con Linux



Práctica

Objetivo

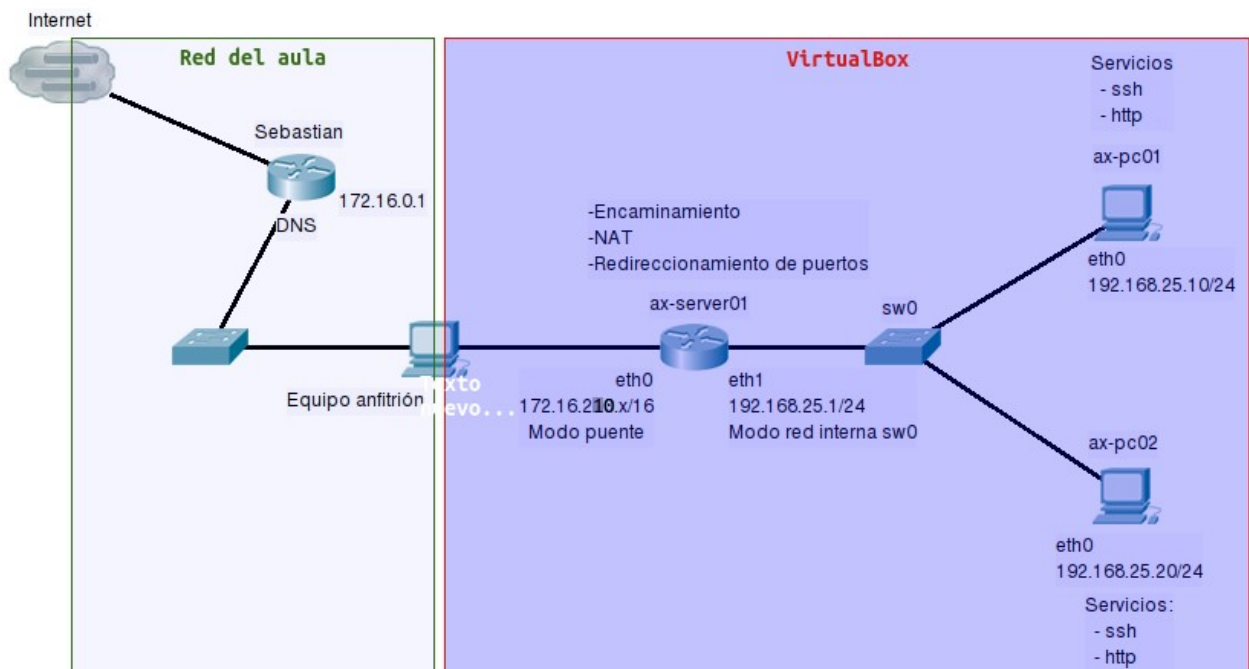
El objetivo es que utilizar la técnica de reenvío de puertos para acceder a servicios que están en una red privada.

Resumen

Para poder realizar la práctica se darán los siguientes pasos genéricos:

- Utilizaremos como base máquinas virtuales de Ubuntu Server, sin entorno gráfico, con el software imprescindible.
- Configurar la red en todas las máquinas virtuales de la misma forma que en dicha práctica, de forma que en el equipo que hace de router tengamos habilitado enrutamiento y el NAT y configurado como puerta de enlace predeterminado el router de clase y las máquinas internas accederán a Internet a través de la máquina virtual que hace de router.
- En ambas máquinas internas instalaremos los servicios ssh y apache2 y en el equipo router configuraremos el reenvío de puertos (PAT) para poder acceder a los servicios de los equipos internos.

Esquema de red de la práctica



Pasos

1) Configuración inicial de las máquinas virtuales.

Configuramos las tarjetas de red en las máquinas virtuales.

- **ax-server01**: 2 tarjetas de red, una en modo puente y la otra en modo red interna conectada al switch virtual sw0.
- **ax-pc01** y **ax-pc02**: 1 tarjeta de red en modo red interna conectada al switch virtual sw0.

Arrancamos las máquinas virtuales y configuramos:

- Nombres de los equipos (ficheros **/etc/hostname** y **/etc/hosts**):
 - **ax-server01**, **ax-pc01** y **ax-pc02**
- Configuración de la IP de las tarjetas de red de los equipos
 - **ax-server01**:
 - tarjeta red modo puente: 172.16.210.x/16. Puerta de enlace 172.16.0.1, DNS 172.16.0.1
 - tarjetas de red modo red interna: 192.168.25.1/24.
 - **ax-pc01**:
 - 192.168.25.10/24. Puerta de enlace 192.168.25.1
 - **ax-pc02**:
 - 192.168.25.20/24 . Puerta de enlace 192.168.25.1
- Activamos reenvío de paquetes en **ax-server01**, ejecutamos:
 - \$ sudo su** ← para convertirnos en root
 - # echo "1" > /proc/sys/net/ipv4/ip_forward** ← activamos reenvío
- Para que los equipos de las **redes internas** puedan acceder a Internet hemos de activar el enmascaramiento (NAT) para que salgan utilizando la IP “pública” del equipo que hacen de router. En **ax-server01** ejecutamos (suponiendo que **enp0s3** es la interfaz externa):

```
$ sudo iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE
```

2) Instalación de servicios en las máquinas internas

Después de comprobar que hemos realizado correctamente los pasos anteriores (desde todos los equipos deberíamos tener acceso a Internet)

```
$ ping google.es ← en los tres equipos
```

Para Instalar los servicios en las máquinas internas:

- **ax-pc01** y **ax-pc02**: ejecutamos
 - \$ sudo apt-get update**
 - \$ sudo apt-get install ssh apache2**

Comprobamos que los servicios se están ejecutando comprobando si sus puertos están a la escucha:

```
$ sudo netstat -lptn
```

3) Activamos el reenvío de puertos

Vamos a reenviar los puertos de **ax-server01** para poder acceder desde la red de clase a los servicios de la red interna virtual de acuerdo a la siguiente tabla:

Servicio	Protocolo	IP externa	Puerto externo	IP interna	Puerto interno
ssh	tcp	172.16.210.x	2200	192.168.25.10	22
http	tcp	172.16.210.x	8080	192.168.25.10	80
ssh	tcp	172.16.210.x	2201	192.168.25.20	22
http	tcp	172.16.210.x	8081	192.168.25.20	80

Para realizar el reenvío de puertos en el equipo **ax-server01**, ejecutamos:

```
$ sudo su
```

```
# iptables -A PREROUTING -t nat -i <enpxsy> -p tcp --dport <puerto_externo> -j  
DNAT --to <IP_interna>:<puerto_interno>
```

Donde:

- **<enpxsy>** ← Interfaz de red externa del equipo, normalmente debería ser enp0s3.
- **<puerto_externo>** ← Puerto de **ax-server01** que se reenviará al servicio de la máquina interna.
- **<IP_interna>:<puerto_interno>** ← Dirección IP de la máquina interna y puerto por el que escucha el servicio de la máquina interna al que queremos dar acceso.

Ejemplo:

Para acceder al servicio ssh de 192.168.25.10 a través del puerto 2200 de **ax-server01** ejecutaríamos:

```
# iptables -A PREROUTING -t nat -i enp0s3 -p tcp --dport 2200 -j DNAT --to  
192.168.25.10:22
```

Ejecuta los comando que permiten reenviar el resto de puertos

4) Accediendo a los servicios desde la red de clase

Para acceder a los servicios internos desde cualquier equipo de la red de clase sólo tenemos que abrir el **cliente de dicho servicio** e introducir la **dirección adecuada**:

- Servidores web internos:
 - Abrimos navegador e introducimos: `http://172.16.210.x:8080` y `http://172.16.210.x:8081`
- Servidores ssh internos
 - Abrimos terminal y ejecutamos: `ssh usuario@172.16.210.x -p 2200` y `ssh usuario@172.16.210.x -p 2201`

5) Guardando los cambios para que se ejecuten al reiniciar el equipo

En el equipo router creamos un script con el nombre nat.sh:

```
$ nano nat.sh
```

Insertamos en el mismo los comandos que ejecutamos para activar el enrutamiento, activar el NAT y realizar la redirección de puertos:

```
echo "1" > /proc/sys/net/ipv4/ip_forward
```

```
iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE
```

```
iptables -A PREROUTING -t nat -i enp0s3 -p tcp --dport 2200 -j DNAT --to 192.168.25.10:22
```

```
iptables -A PREROUTING -t nat -i enp0s3 -p tcp --dport 8080 -j DNAT --to 192.168.25.10:80
```

```
iptables -A PREROUTING -t nat -i enp0s3 -p tcp --dport 2201 -j DNAT --to 192.168.25.20:22
```

```
iptables -A PREROUTING -t nat -i enp0s3 -p tcp --dport 8081 -j DNAT --to 192.168.25.20:80
```

Guardamos el fichero y luego lo copiamos a la carpeta en la que se almacenan los scripts de inicio:

```
$ sudo mv nat.sh /etc/init.d
```

Le damos permiso de ejecución:

```
$ sudo chmod +x /etc/init.d/nat.sh
```

Hacemos que se ejecute al inicio del sistema insertando el fichero en `/etc/rc.local`

```
$ sudo nano /etc/rc.local
```

Añadimos en la penúltima línea, antes de `exit 0`

```
$ /etc/init.d/nat.sh
```

Si ahora reinicamos el equipo router se debería mantener el enrutamiento, el NAT y el redireccionamiento de puertos.

Cuando termines avisa al profesor para que revise la práctica