

Enrutamiento Estático

Los routers son dispositivos de red que saben comunicarse con cualquier red que tengan conectada directamente. Si por ejemplo tenemos un equipo ubuntu llamado **PC1** con dos tarjetas de red:

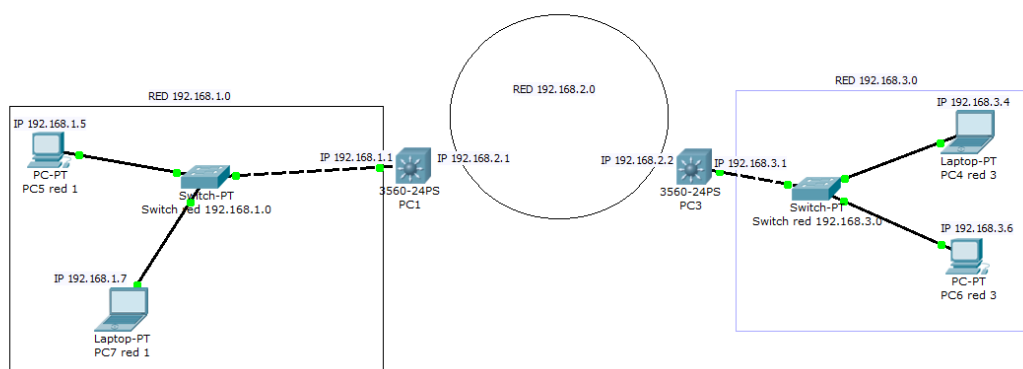
- eth0 con IP/MASK: 192.168.1.1/24
- eth1 con IP/MASK: 192.168.2.1/24

Puede comunicarse con las redes 192.168.1.0 y la 192.168.2.0 sin problemas. Pero no puede comunicarse con la red 192.168.3.0/24 porque no la tiene conectada.

Imaginemos ahora otro equipo llamado **PC3** con dos tarjetas de red:

- eth0 con IP/MASK: 192.168.2.2/24
- eth1 con IP/MASK 192.168.3.1/24

Por lo que hemos dicho antes este segundo equipo puede comunicarse sin problemas con la red 192.168.3.0 y con la 192.168.2.0



Los equipos **PC1** y el **PC3** tienen ambos una tarjeta de red conectada a la red **192.168.2.0/24** así que pueden hablar entre ellos. Eso significa que si el **PC1** le envía los paquetes de la **red 3** al **PC3**, éste último podría entregarlos ya que está conectado a esa red. Y viceversa, el **PC3** si le envía los paquetes de la **red 1** al **PC1** éste podría entregarlos.

Lo que tenemos que hacer es “decirle” al PC1 que cuando quiera enviar paquetes a la red 3 que “hable” con el PC3

También tenemos que poder hacer el **camino a la inversa**, para ello hay que decirle al PC3 que para enviar paquetes a la red 1 tiene que “hablar” con el PC1

La forma de decirle a un equipo que debe “hablar” con otro para enviar paquetes es con el comando route.

En el PC1 pondríamos:

```
# route add -net 192.168.3.0 netmask 255.255.255.0 gw 192.168.2.2
```

con este comando le estamos diciendo que para **alcanzar la red**(net) 192.168.3.0 debe usar la **puerta de enlace** (gw) **192.168.2.2** que como ya sabemos es la IP del PC3 conectada a la red en la que coincide con PC1.

En el PC3 habría que poner:

```
# route add -net 192.168.1.0 netmask 255.255.255.0 gw 192.168.2.1
```

y con esto ya tenemos un “camino” entre las dos redes:

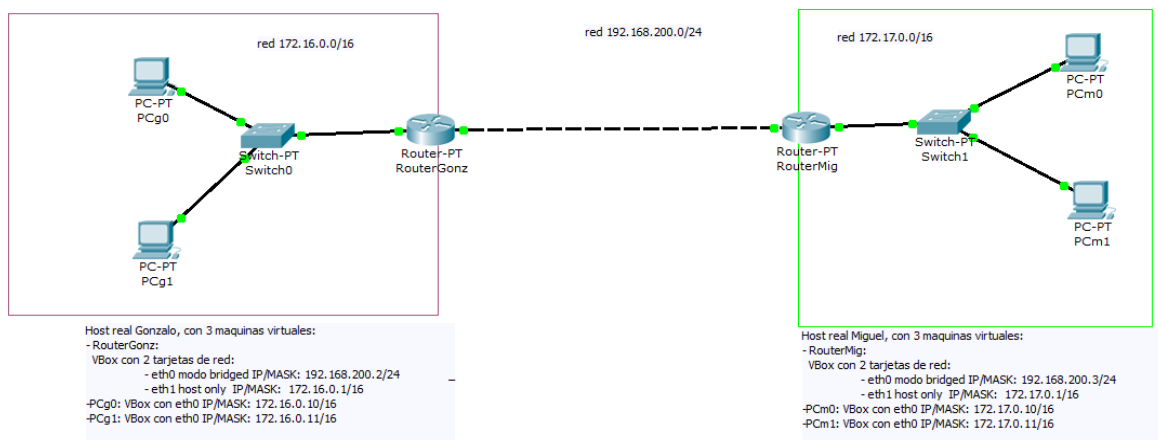
PRÁCTICA (A realizar en parejas):

Vamos a conectar mediante enrutamiento estático dos redes simuladas mediante VirtualBox en los equipos servidor y cliente que estamos utilizando actualmente. Para ello en un Host real que llamaremos Gonzalo pondremos 3 máquinas virtuales:

- equipo PCg0 con eth0 modo red interna (conectado a swg1, por ejemplo), IP 172.16.0.10 y máscara 255.255.0.0
- equipo PCg1 con eth0 modo red interna (conectado a swg1), IP 172.16.0.11 y máscara 255.255.0.0
- equipo RouterGonz con dos tarjetas de red y dos Ips:
 - eth0 modo bridge (a la tarjeta de red que me da acceso al equipo de Miguel) con IP 192.168.200.2 y máscara 255.255.255.0
 - eth1 modo red interna (conectado a swg1) con IP 172.16.0.1

Ahora en otro host real que llamaremos Miguel otras 3 máquinas virtuales:

- equipo PCm0 con eth0 modo red interna (conectado a swm1, por ejemplo), IP 172.17.0.10 y máscara 255.255.0.0
- equipo PCm1 con eth0 modo red interna (conectado a swm1), IP 172.17.0.11 y máscara 255.255.0.0
- equipo RouterMig con dos tarjetas de red y dos Ips:



- eth0 modo bridge con IP 192.168.200.3 y máscara 255.255.255.0
- eth1 modo red interna (conectado a swm1) con IP 172.17.0.1

Para instalar VirtualBox en Ubuntu Server tenemos que realizar los siguientes pasos:

```
$ sudo apt-get install xfce4      ← instalamos un entorno de ventanas simple
$ sudo apt-get install virtualbox
$ startx                          ← iniciamos el modo gráfico
```

Vamos a configurar manualmente la IP, máscara y enrutamiento de cada uno de los equipos. Para configurar las Ips y máscara de cada tarjeta de red en cada equipo ejecutamos:

```
# ifconfig ethx IP netmask mascara up
```

Donde: **ethx** - es la interfaz que vamos a configurar (eth0, eth1), **IP** - es la dirección que le vamos a asignar a la interfaz (172.16.0.1), **mascara** - la máscara de red (255.255.0.0) y **up** - aplica la configuración y activa la interfaz

Tanto RouterGonz y RouterMig deben tener activo el bit de forwarding para que puedan enrutar

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

La gateway (puerta de enlace) por defecto de PCg0 y PCg1 será la IP de RouterGonz:

```
# route add default gw 172.16.0.1
```

La gateway (puerta de enlace) por defecto de PCm0 y PCm1 será la IP de RouterMig:

```
# route add default gw 172.17.0.1
```

Le decimos a RouterGonz que para llegar a los equipos de la red de Miguel tenemos que hablar con RouterMig:

```
# route add -net 172.17.0.0 netmask 255.255.0.0 gw 192.168.200.3
```

Finalmente decir a RouterMig que para llegar a los equipos de la red de Gonzalo tenemos que hablar con RouterGonz:

```
# route add -net 172.16.0.0 netmask 255.255.0.0 gw 192.168.200.2
```

Para ver la tabla de rutas que tenemos creada usamos el comando route.

```
# route -n
```

Comprueba que todos los equipos son accesibles y muestra la tabla de enrutamiento de los equipos Router y PC y explica para qué sirve cada línea