

Crear tablas de ruta a partir de esquemas

Direccionamiento ip

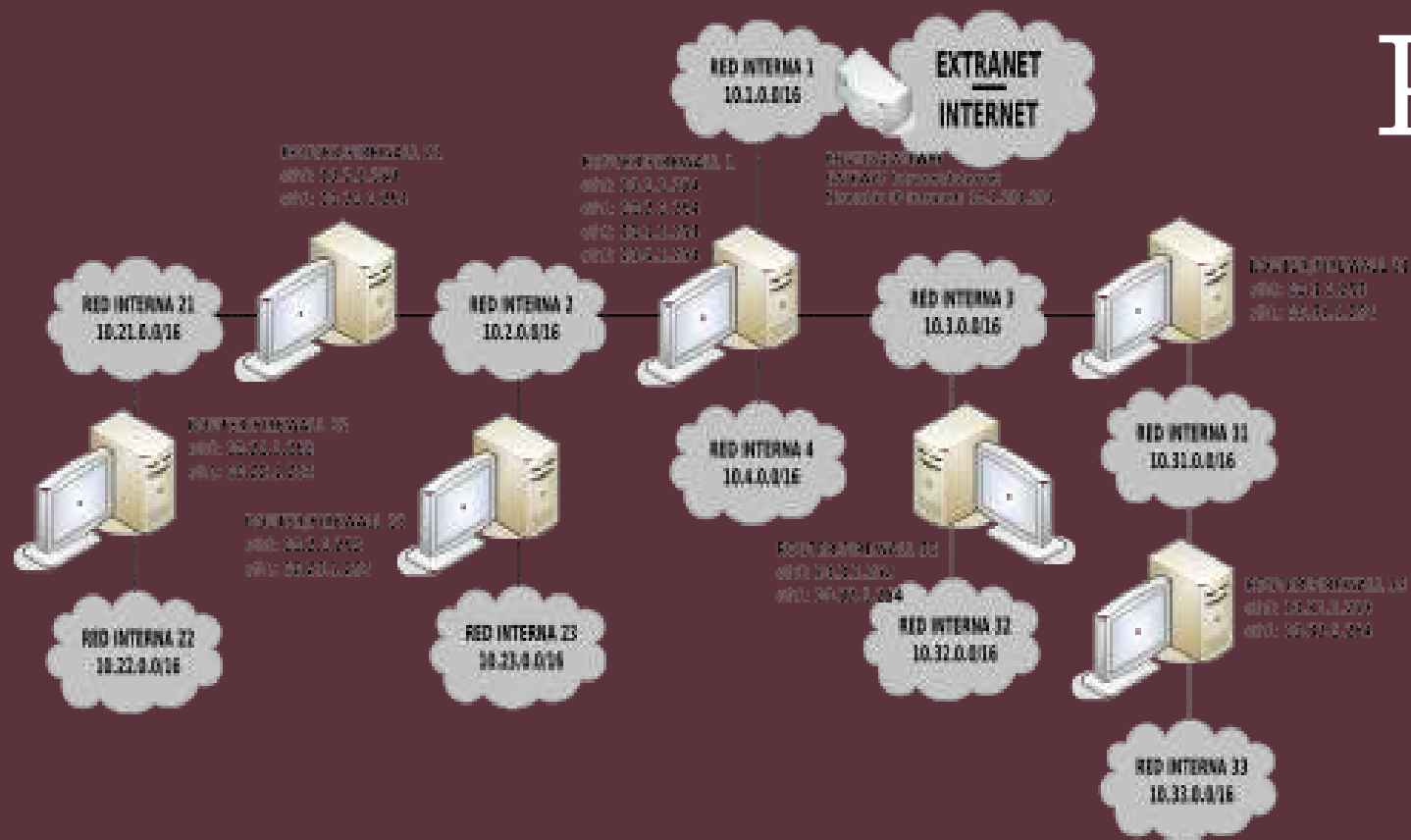


Práctico 1.2.8: Esquema de Red a partir de la Tabla de Enrutamiento I

Observa el esquema de red que muestra a continuación y responde a las siguientes cuestiones prácticas:

- A. ¿Cuál sería la tabla de enrutamiento del router N°1, dando por hecho que en su tabla de enrutamiento existen reglas para alcanzar cualquiera de las redes del esquema?
- B. Para saber más....¿Cuál sería el contenido del script de configuración que debería ejecutarse en la máquina router N°1 al iniciarse para que se autoconfigurara correctamente como gateway de la red.

Esquema de red:



Solución Ej. Pr. 1.2.8.- Tablas de enrutamiento y Script de Configuración de Red

Tras observar el esquema de red, deberíamos concluir que la tabla de enrutamiento del router N°1 debería ser la siguiente:

Una posible solución

Redes Destino	Máscara	Gateway	Interfaz Red
10.1.0.0	255.255.0.0	0.0.0.0	eth0
10.2.0.0	255.255.0.0	0.0.0.0	eth1
10.3.0.0	255.255.0.0	0.0.0.0	eth2
10.4.0.0	255.255.0.0	0.0.0.0	eth3
10.21.0.0	255.255.0.0	10.2.1.253	eth1
10.22.0.0	255.255.0.0	10.2.1.253	eth1
10.23.0.0	255.255.0.0	10.2.1.252	eth1
10.32.0.0	255.255.0.0	10.3.1.252	eth2
10.31.0.0	255.255.0.0	10.3.1.253	eth2
10.33.0.0	255.255.0.0	10.3.1.253	eth2
0.0.0.0	0.0.0.0	10.1.254.254	eth0



Para saber más....

- Para la configurar la tabla de enrutamiento anterior en el equipo gateway podremos hacer uso de (1) los scripts de configuración disponibles en el sistema o (2) crear nuestro propio script de configuración:

) En el caso de querer hacer uso de los scripts ya proporcionados por el sistema haríamos los siguiente

a) Editaríamos el archivo /etc/network/interfaces para asignar las direcciones IP a las interfaces de red del equipo, indicar quien es su gateway por defecto, quienes son sus servidores DNS y para añadir las reglas de enrutamiento estático que le informen de como alcanzar todas las redes del entorno:

```
[root@linux]# nano /etc/network/interfaces
# Contenido del archivo interfaces:
auto lo eth0 eth1 eth2 eth3 iface
lo inet loopback

iface eth0 inet static
address 10.1.1.254
netmask 255.255.0.0
gateway 10.1.254.254 dns-
nameservers 8.8.8.8

iface eth1 inet static address 10.2.1.254 netmask 255.255.0.0 post-up
route add -net 10.21.0.0 netmask 255.255.0.0 gw 10.2.1.253 post-up
route add -net 10.22.0.0 netmask 255.255.0.0 gw 10.2.1.253 post-up
route add -net 10.23.0.0 netmask 255.255.0.0 gw 10.2.1.252 pre-
down route del -net 10.21.0.0 netmask 255.255.0.0 gw 10.2.1.253
pre-down route del -net 10.22.0.0 netmask 255.255.0.0 gw
10.2.1.253 pre-down route del -net 10.23.0.0 netmask 255.255.0.0
gw 10.2.1.252

iface eth2 inet static address 10.3.1.254 netmask 255.255.0.0 post-up
route add -net 10.31.0.0 netmask 255.255.0.0 gw 10.3.1.253 post-up
route add -net 10.32.0.0 netmask 255.255.0.0 gw 10.3.1.252 post-up
route add -net 10.33.0.0 netmask 255.255.0.0 gw 10.3.1.253 pre-
down route del -net 10.31.0.0 netmask 255.255.0.0 gw 10.3.1.253
pre-down route del -net 10.32.0.0 netmask 255.255.0.0 gw
10.3.1.252 pre-down route del -net 10.33.0.0 netmask 255.255.0.0
gw 10.3.1.253

iface eth3 inet static
address 10.4.1.254
netmask 255.255.0.0
```

Para que surta efecto la configuración anterior deberemos reactivar las interfaces de red o directamente reiniciar el equipo (*init 0*):

```
[root@linux]# ifdown -a  
[root@linux]# ifup -a
```

Estas instrucciones están
sustituidas por `itconfig eth0
up/down`

- a) Tras la configuración IP tan sólo deberemos activar el `ip_forwarding` para permitir el reenvío de paquetes entre las diferentes interfaces de red, además de activar el enmascaramiento de los paquetes que le atraviesen con la finalidad de que los paquetes de retorno puedan llegar a su destino (*cambia la dirección IP de origen del equipo de la intranet por la dirección IP de la interfaz de salida del router*). Una posibilidad sería añadir los comandos necesarios al script

```
[root@linux]# echo 'echo "1" > /proc/sys/net/ipv4/ip_forward' >> /etc/init.d/rc.local
[root@linux]# echo 'iptables -t nat -A POSTROUTING -j MASQUERADE' >> /etc/init.d/rc.local
```

`/etc/init.d/rc.local`, el cual se ejecuta al iniciarse la máquina:

- En relación al script de configuración, tal como se ha mostrado en ejercicios anteriores, debería tener el siguiente aspecto:

```
[root@linux]# nano /etc/init.d/conf-red
```

```
#!/bin/bash
```

```
/etc/init.d/networking stop ifconfig eth0 10.1.1.254 netmask  
255.255.0.0 ifconfig eth1 10.2.1.254 netmask 255.255.0.0 ifconfig  
eth2 10.3.1.254 netmask 255.255.0.0 ifconfig eth3 10.4.1.254 netmask  
255.255.0.0 route add -net 10.21.0.0 netmask 255.255.0.0 gw  
10.2.1.253 dev eth1 route add -net 10.22.0.0 netmask 255.255.0.0 gw  
10.2.1.253 dev eth1 route add -net 10.23.0.0 netmask 255.255.0.0 gw  
10.2.1.252 dev eth1 route add -net 10.31.0.0 netmask 255.255.0.0 gw  
10.3.1.253 dev eth2 route add -net 10.32.0.0 netmask 255.255.0.0 gw  
10.3.1.252 dev eth2 route add -net 10.33.0.0 netmask 255.255.0.0 gw  
10.3.1.253 dev eth2 route add default gw 10.1.254.254 echo  
"nameserver 8.8.8.8" > /etc/resolv.conf  
echo "1" > /proc/sys/net/ipv4/ip_forward  
iptables -t nat -A POSTROUTING -j MASQUERADE
```