UNIVERSIDAD DE CANTABRIA

DEPARTAMENTO DE INGENIERÍA DE COMUNICACIONES

GRUPO DE INGENIERÍA TELEMÁTICA

PROTOCOLOS PARA LA INTERCONEXIÓN DE REDES
PRÁCTICA 1 – CONFIGURACIÓN Y ANÁLISIS DE REDES TCP/IP

Práctica 1: Configuración y análisis de redes

Objetivos:
Durante esta práctica se van a reforzar los conocimientos adquiridos por los alumnos en las clases de teoría a través de la configuración de redes, el análisis y monitorización de su funcionamiento y la resolución de los fallos detectados a través de este análisis.

Para ello, se van a realizar las siguientes acciones:

- Configuración de los parámetros básicos de una red (implementada sobre máquinas Linux).
- Análisis del funcionamiento de la red y resolución de los posibles fallos en la configuración.
- Análisis del tráfico TCP-UDP/IP sobre la red mediante el analizador de protocolos.

Al finalizar la práctica, cada grupo deberá completar un documento con las respuestas a todas las preguntas realizadas a lo largo de la práctica. Este documento, se enviará por correo electrónico a la dirección sanchezgl@unican.es

Introducción:
Los conocimientos adquiridos durante las clases teóricas se pondrán a prueba en esta práctica en la que el alumno asumirá el rol de administrador de red ya que tendrá que proceder a la configuración de los diferentes dispositivos y routers de una red.

Para ello, se empleará una herramienta de emulación de redes en la cual los diferentes elementos de la red (dispositivos de usuario y routers) se implementan a través de máquinas virtuales Linux. Netkit [1] es un entorno software que permite realizar experimentos con redes de ordenadores virtuales sin necesidad de disponer de dispositivos de comunicaciones ni ordenadores reales. Netkit permite interconectar varios nodos virtuales (ordenadores, hubs y routers) que emulan el funcionamiento de nodos con el S.O. GNU/Linux. Por su parte NetGUI [2] es una interfaz gráfica para el sistema Netkit que permite una interacción más amigable e intuitiva.

En esta práctica se procederá a crear varias redes usando el entorno de NetGUI (Ver Anexo 3) para posteriormente mediante el uso de las herramientas de configuración y análisis de la red que implementa Linux (Ver Anexos 1 y 2) monitorizar su funcionamiento y el de los protocolos implicados.

Creación de un diagrama de red:

1. Arranca NetGUI. En el aula de prácticas la forma de hacerlo es ejecutando en una ventana de terminal la orden `sudo netgui.sh`

2. Crea una red como la de la Figura 1, en donde *pc1*, *pc2* son 2 ordenadores, y *r1* es un router.

Para conectar varios dispositivos a un mismo segmento de red necesitarás utilizar hubs:

- *hub1* para conectar *pc1* y *r1*.
- *hub2* para conectar *pc2* y *r1*.

Fíjate en el orden en el que dibujas los cables para que las interfaces de los routers se enumeren de la misma forma que aparecen en el dibujo. En el caso de *r1* primero deberías dibujar el cable que une a *r1* con *hub1* y después el que une *r1* con *hub2*. De esta forma las interfaces *eth0* y *eth1* quedarán enumeradas de la misma manera que en el dibujo.

Cuando hayas terminado de dibujar la red, el aspecto que tendrá en NetGUI debería parecerse a la Figura 1. (en NetGUI las direcciones IP no aparecen hasta que no se arrancan las máquinas).

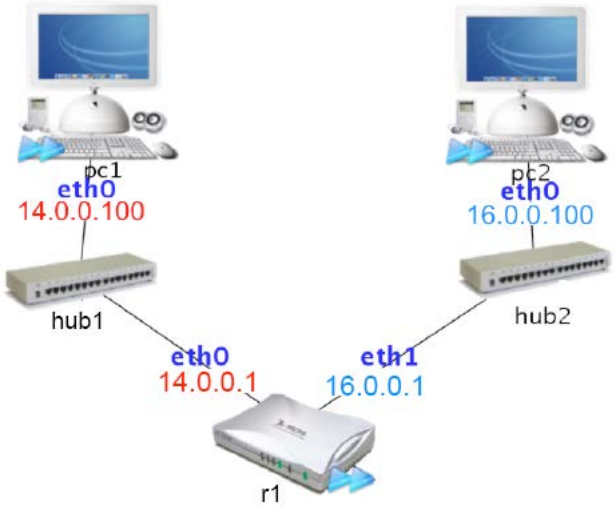


Figura 1. Diagrama de red básico

3. Arranca los ordenadores y el router de uno en uno, y comprueba la configuración de la red en cada uno de ellos mediante la orden `ifconfig`.

4. Las máquinas recién arrancadas no tienen configuradas sus interfaces de red Ethernet por lo que ninguna aplicación podrá intercambiar mensajes con otras máquinas. Sólo podrá hacerlo cada máquina consigo misma a través de la dirección 127.0.0.1. La dirección IP 127.0.0.1 está asignada a la interfaz de loopback *lo*. La interfaz de loopback es virtual: los datagramas que se envían a través de la interfaz de loopback vuelven a la misma máquina sin llegar a salir a la red. Escribe en *pc1* la orden `ping 127.0.0.1`. Comprueba cómo la máquina *pc1* responde a sus propios pings destinados a la dirección 127.0.0.1.

5. Comprueba la tabla de rutas de las máquinas. Verás que está vacía en todas ellas.

6. Utilizando las órdenes `ifconfig` o `ip`, asigna las siguientes direcciones IP a las interfaces de red de las máquinas de la siguiente forma:

pc1 (eth0): 14.0.0.100

pc2 (eth0): 16.0.0.100

r1 (eth0): 14.0.0.1

r1 (eth1): 16.0.0.1

La máscara de subred debe ser en todos los casos la 255.255.255.0.

Comprueba que cada interfaz tiene la dirección IP adecuada en cada máquina.

7. Con el comando `route` puedes comprobar cómo tras asignar la dirección IP a una interfaz de red de una máquina se añade automáticamente una entrada en la tabla de encaminamiento de la máquina. ¿Qué nos permiten las nuevas entradas que han aparecido?

8. Comprueba cómo ahora sí funcionan los `ping` a direcciones de la misma red. Haz ping a 14.0.0.1 desde *pc1* y ping a la dirección 16.0.0.1 desde *pc2*.

9. Comprueba lo que ocurre cuando haces un ping desde *pc1* a 16.0.0.1. Antes de ejecutar el comando y teniendo en cuenta que desde *pc1* ya se ha hecho un ping a *r1* sin problemas, ¿crees que funcionará? ¿Por qué?

10. Añade una ruta por defecto en *pc1* para que los datagramas IP que no sean para su propia red los envíe a su router por defecto (en este caso *r1*). ¿Cuál será la dirección IP de *r1* que habrá que poner en el parámetro `gw` del comando `route`?

Comprueba que ahora el ping a 16.0.0.1 desde *pc1* sí funciona.

11. Sin embargo, sigue sin funcionar el ping de *pc1* a *pc2*. ¿Por qué?

Arréglalo y comprueba que funciona el ping de *pc1* a *pc2*. Si a priori no se te ocurre cual es el problema, captura y analiza el tráfico en ambas subredes para detectar que es lo que está ocurriendo.

12. Utilizando las herramientas `tcpdump` y `wireshark`, responde a las siguientes preguntas sobre el escenario anterior.

Para responder adecuadamente deberás decidir en qué momento y en qué subredes tienes que arrancar `tcpdump`.

Haz un ping de *pc1* a *pc2* para que envíe 3 paquetes ICMP:

¿Con qué TTL llegan los mensajes ICMP a *pc2*?

¿Qué valor tienen los campos Type y Code de los mensajes ICMP que llegan a *pc2*?

¿Qué valor tienen los campos Type y Code de los mensajes ICMP que llegan a *pc1*?

¿Qué valor numérico hexadecimal tiene el campo Protocol de los datagramas IP en los que viajan los mensajes ICMP?

¿Qué valor numérico hexadecimal tiene el campo de tipo de protocolo de las tramas Ethernet en las que viajan los mensajes ICMP?

¿En qué subred/subredes has tenido que capturar el tráfico con tcpdump para responder a las preguntas de este apartado?

Diagramas de red pre-generados: Escenarios de red:
En NetGUI es posible cargar redes previamente guardadas y trabajar sobre ellas.

Escenario A

Cierra NetGUI si está lanzado, y vuelve a lanzarlo. En el menú, elige Archivo->Abrir y elige el directorio *lab-p0a* (incluido en */home/alumnos/Asignaturas/PIR/PR1*). Arranca todas las máquinas de dicho escenario, de una en una, esperando que una máquina haya terminado su arranque antes de arrancar la siguiente. Obtendrás un escenario como el que se muestra en la Figura 2. Verás, que cuando terminan de arrancar, algunas de las máquinas tienen ya configurada su dirección IP y algunas rutas.

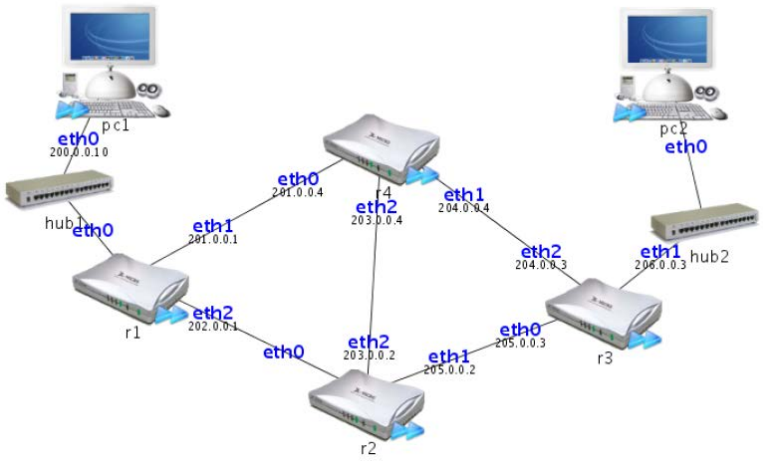


Figura 2. Escenario de red lab-p0a

Algunas máquinas del escenario lab-p0a necesitan configurar direcciones IP y/o rutas. Irás realizando dicha configuración a lo largo de los siguientes ejercicios.

Parte 1:

En *pc1* se ejecuta el siguiente comando:

```
pc1:~# traceroute 203.0.0.4
traceroute to 203.0.0.4 (203.0.0.4), 64 hops max, 40 byte packets
1 200.0.0.1
2 202.0.0.2
3 203.0.0.4
```

En *r4* se ejecuta el siguiente comando:

```
r4:~# traceroute 200.0.0.10
traceroute to 200.0.0.10 (200.0.0.10), 64 hops max, 40 byte packets
1 203.0.0.2
2 202.0.0.1
3 200.0.0.10
```

1. Realiza los cambios de configuración necesarios para que el resultado anterior sea posible. Efectúa sólo los cambios imprescindibles. No modifiques las rutas ni las direcciones IP que ya están configuradas en el escenario, sólo puedes añadir direcciones IP y rutas. En las tablas de encaminamiento de los ordenadores sólo puedes añadir rutas

por defecto. En las tablas de encaminamiento de los routers NO puedes añadir rutas por defecto.

2. ¿Cuáles son los routers que se atraviesan para ir desde *pc1* a la dirección 203.0.0.4?

3. ¿Cuáles son los routers que se atraviesan para ir desde *r4* a *pc1*?

4. Comprueba la configuración que has realizado ejecutando el `traceroute` anterior.

5. Anota las rutas que has añadido en los diferentes routers de la red:

r1:

Destino	Gateway	Máscara	Interfaz

r2:

Destino	Gateway	Máscara	Interfaz

r3:

Destino	Gateway	Máscara	Interfaz

r4:

Destino	Gateway	Máscara	Interfaz

Parte 2:

En *pc1* realiza un traceroute a *pc2* y se obtiene el siguiente resultado:

```
pc1:~# traceroute 206.0.0.10
traceroute to 206.0.0.10 (206.0.0.10), 64 hops max, 40 byte packets
1 200.0.0.1
2 202.0.0.2
3 204.0.0.3
4 206.0.0.10
```

1. Realiza los cambios de configuración necesarios para que el resultado anterior sea posible. Efectúa sólo los cambios imprescindibles. No modifiques las rutas ni las direcciones IP que ya están configuradas en el escenario, sólo puedes añadir direcciones IP y rutas. En los ordenadores sólo puedes añadir rutas por defecto. En los routers NO puedes añadir rutas por defecto.

2. ¿Cuáles son los routers que se atraviesan para ir desde *pc1* a *pc2*?

3. Explica por qué en el resultado de `traceroute` la dirección IP del tercer salto es 204.0.0.3 en vez de 205.0.0.3.

4. Comprueba la configuración que has realizado ejecutando el `traceroute` anterior.

5. Anota las rutas que has añadido en los diferentes routers de la red:

r1:

Destino	Gateway	Máscara	Interfaz

r2:

Destino	Gateway	Máscara	Interfaz

r3:

Destino	Gateway	Máscara	Interfaz

r4:

Destino	Gateway	Máscara	Interfaz

Parte 3:

En *pc2* se ha realizado un `traceroute` a la dirección 201.0.0.1 y se ha obtenido en *pc2* la captura dada en el fichero cap1.cap (lo podéis encontrar en la carpeta */home/alumnos/Asignaturas/PIR/PR1* de la máquina real).

1. Realiza los cambios de configuración necesarios para que la captura proporcionada sea posible. Efectúa sólo los cambios imprescindibles. No modifiques las rutas ni las direcciones IP que ya están configuradas en el escenario, sólo puedes añadir direcciones IP y rutas. En los ordenadores sólo puedes añadir rutas por defecto. En los routers NO puedes añadir rutas por defecto.

2. ¿Cuáles son los routers que se atraviesan para ir desde *pc2* a 201.0.0.1?

3. Observando el valor del campo TTL en el paquete número 10 de la captura, ¿es posible que el camino desde *pc2* a 201.0.0.1 sea el mismo que el camino desde 201.0.0.1 a *pc2*? ¿Qué otra información obtenida en el fichero de captura justifica tu respuesta?

4. Comprueba la configuración que has realizado ejecutando el `traceroute` anterior

Escenario B

Cierra NetGUI si está lanzado, y vuelve a lanzarlo. En el menú, elige Archivo->Abrir y elige el directorio lab-p0b (incluido en */home/alumnos/Asignaturas/PIR/PR1*). Arranca todas las máquinas de dicho escenario, de una en una, esperando que una máquina haya terminado su arranque antes de arrancar la siguiente. Obtendrás un escenario como el que se muestra en la Figura 3.

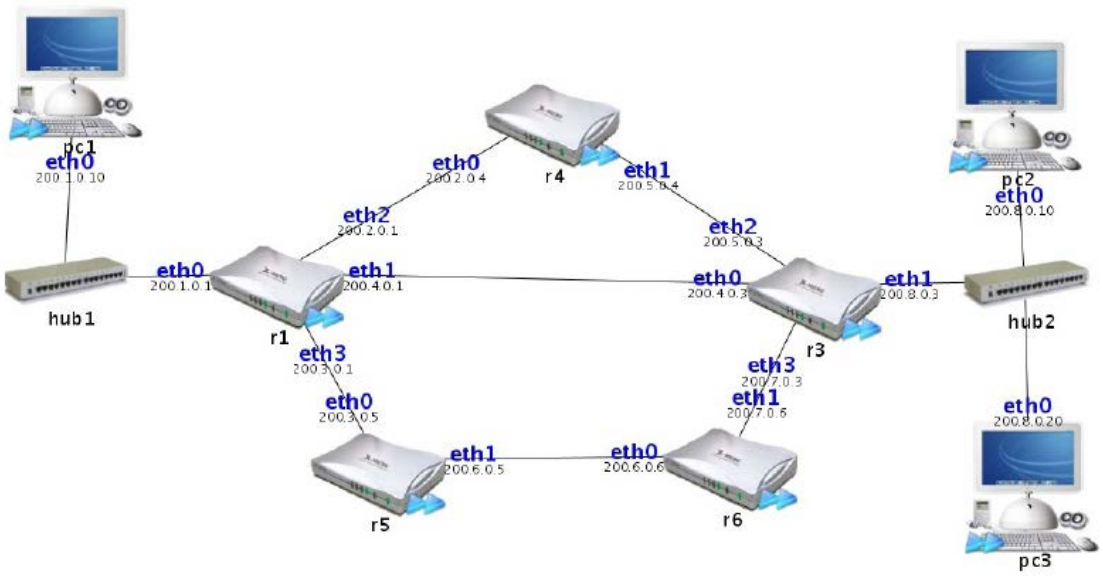


Figura 3. Escenario de red lab-p0b

Como comprobarás, el escenario que se proporciona no está configurado completamente. Algunas máquinas necesitan configurar rutas. Irás realizando dicha configuración a lo largo de la práctica.

Parte 1:

En el escenario anterior se ejecutan una o varias instrucciones y se realizan las siguientes capturas:

- cap2.cap: Captura realizada en la red 200.6.0.0.
- cap3.cap: Captura realizada en la red 200.4.0.0.

1. Configura todas las rutas que consideres necesarias para que estas capturas sean posibles.

2. Indica qué instrucción/es se han ejecutado y donde lo has hecho para obtener el tráfico de estas capturas.

3. Comprueba la configuración que has realizado obteniendo de nuevo estas capturas.

Parte 2:

En el escenario anterior se ejecutan una o varias instrucciones y se realiza la siguiente captura: cap4.cap.

1. Configura todas la rutas que consideres necesarias para que esta captura sea posible.

2. Indica en qué subred se ha realizado la captura.

3. Indica qué instrucción/es se han ejecutado y donde lo has hecho para obtener el tráfico de esa captura.

4. Comprueba la configuración que has realizado obteniendo de nuevo esta captura.

Anexo 1: Herramientas de configuración de red en Linux

Linux permite la configuración de diferentes parámetros de nivel de red (principalmente direcciones IP de los interfaces y entradas de las tablas de enrutamiento) a través de comandos del sistema.

A través de los comandos:

- `ifconfig`
- `ip`
- `route`

el administrador del sistema puede añadir / eliminar / modificar las direcciones IP asignadas a cada uno de los interfaces de red de un determinado equipo así como introducir / eliminar rutas en las tablas de encaminamiento de los dispositivos.

A continuación se presenta el modo en el que estas acciones se pueden llevar a cabo:

Mostrar información de las interfaces de red

- Con `ifconfig`:

```
pc1:~# ifconfig
eth0  Link encap:Ethernet Hwaddr 0A:29:92:55:93:70
      inet addr:10.0.0.1 Bcast:10.0.0.255 Mask: 255.255.255.0
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
      RX packets:4 errors:0 dropped:0 overruns:0 frame:0
      TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:224 (224.0 b) TX bytes:280 (280.0 b)
      Interrupt:5
lo    Link encap:Locap Loopback
      inet addr:127.0.0.1 Bcast:255.0.0.0
      UP LOOPBACK RUNNING MTU:16436 Metric:1
      RX packets:6 errors:0 dropped:0 overruns:0 frame:0
      TX packets:6 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:0
      RX bytes:504 (504.0 b) TX bytes:504 (504.0 b)
```

- Con `ip`:

```
pc1:~# ip address show
0: lo: <LOOPBACK,UP,10000> mtu 16436 qdisc noqueue
     link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
     inet 127.0.0.1/8 scope host lo
1: eth0: <BROADCAST,MULTICAST,UP,10000> mtu 1500 qdisc pfifo_fast qlen 1000
     link/ether 0A:29:92:55:93:70 brd ff:ff:ff:ff:ff:ff
     inet 10.0.0.1/24 brd 10.0.0.255 scope global eth0
```

Añadir / Eliminar una dirección IP

Para configurar una dirección IP es necesario saber sobre que interfaz de red vamos a trabajar, así como la propia dirección IP que queremos asignarle y la máscara de red que se va a usar.

- Añadir una dirección IP:

```
ifconfig <interfaz> <dirIP> netmask <máscara>
  pc1:~# ifconfig eth0 10.0.0.1 netmask 255.255.255.0
ip address add dev <interfaz> <dirIP/prefijoMáscara> broadcast +
  pc1:~# ip link set eth0 up
  pc1:~# ip address add dev eth0 10.0.0.1/24 broadcast +
```

- Eliminar una dirección IP:

```
ip address del dev <interfaz> <dirIP/prefijoMáscara>
  pc1:~# ip address del dev eth0 10.0.0.1/24
```

Con `ifconfig` sólo se puede "apagar" la interfaz, que no es exactamente lo mismo que eliminar la dirección IP:

```
  ifconfig <interfaz> down
  pc1:~# ifconfig eth0 down
```

Después de añadir o eliminar una dirección IP es conveniente cerciorarse de que la configuración se ha realizado correctamente mostrando la información de las interfaces de red.

Los cambios realizados de esta manera no se guardan al reiniciar el equipo.

Mostrar la tabla de encaminamiento:

- Con `route`:

```
pc1:~# route -n
Kernel IP routing table
Destination  Gateway  Genmask      Flags        Metric   Ref  Use  Iface
10.0.0.0     *        255.255.255.0    U        0        0    0    eth0
```

- Con `ip`:

```
pc1:~# ip route show
10.0.0.0/24 dev eth0    proto kernel       scope     link    src 10.0.0.1
```

- Con `netstat`:

```
pc1:~# netstat -nr
Kernel IP routing table
Destination  Gateway  Genmask          Flags    MSS  Window   irtt   Iface
10.0.0.0     *        255.255.255.0    U        0    0        0      eth0
```

La opción –n en `route` y `netstat` fuerza a que no se haga la resolución de nombres de las direcciones IP por lo que hace que la ejecución del comando sea más rápida.

Añadir una ruta en la tabla de encaminamiento:

- Con `route`:
    - Ruta a una máquina:

```
route add -host <máquinaDestino> gw <gateway> dev <interfaz>
pc1:~# route add -host 11.0.0.1 gw 10.0.0.1 dev eth0
```

    - Ruta a una red:

```
route add -net <subredDestino> netmask <máscara> gw <gateway> dev <interfaz>
pc1:~# route add -net 12.0.0.0 netmask 255.255.255.0 gw 10.0.0.1 dev eth0
```

    - Ruta por defecto

```
route add default gw <gateway> dev <interfaz>
pc1:~# route add default gw 10.0.0.2 dev eth0
```

El parámetro `gw <gateway>` se refiere al router que debe ser el próximo salto en esa ruta.

El parámetro `dev <interfaz>` se refiere a la interfaz de red por la que se transmitirán los datagramas asociados a esa ruta. Si se incluye el parámetro `gw`, es posible omitir cual será la interfaz ya que se infiere al saberse cual es el router al que hay que enviar los datagramas.

- Con `ip`:
    - Ruta a una máquina o a una red:

```
ip route add <dirP/máscara> via <gateway> dev <interfaz>
pc1:~# ip route add 12.0.0.0/24 via 10.0.0.1 dev eth0
```

    - Ruta por defecto

```
ip route add default via <gateway> dev <interfaz>
pc1:~# ip route add default  via 10.0.0.2 dev eth0
```

Los cambios realizados de esta manera no se guardan al reiniciar el equipo.

Borrar una ruta en la tabla de encaminamiento:

- Con `route`:
    - Ruta a una máquina:

```
route del -host <máquinaDestino>
pc1:~# route del -host 11.0.0.1
```

    - Ruta a una red:

```
route del -net <subredDestino> netmask <máscara>
pc1:~# route del -net 12.0.0.0 netmask 255.255.255.0
```

    - Ruta por defecto

```
route del default
pc1:~# route del default
```

- Con `ip`:
    - Ruta a una máquina o a una red:

```
ip route del <dirP/máscara> via <gateway> dev <interfaz>
pc1:~# ip route del 12.0.0.0/24 via 10.0.0.1 dev eth0
```

    - Ruta por defecto

```
ip route del default via <gateway> dev <interfaz>
pc1:~# ip route del default  via 10.0.0.2 dev eth0
```

Los cambios realizados de esta manera no se guardan al reiniciar el equipo.

Anexo 2: Herramientas de diagnóstico de red en Linux

Linux permite monitorizar el estado de la conectividad entre las máquinas de una red a través de comandos del sistema.

A través de los comandos:

- `arp`
- `ping`
- `traceroute`
- `netstat`
- `tcpdump`

el administrador del sistema puede analizar el estado de una red, monitorizar su tráfico y visualizar los parámetros de red en los distintos dispositivos.

A continuación se presenta el modo en el que estas acciones se pueden llevar a cabo:

Mostrar la caché de ARP

La caché de ARP es la tabla donde se almacenan de manera temporal las asociaciones dirección IP – dirección física que resultan de la resolución de direcciones realizada a través del protocolo ARP. Si una determinada asociación aparece en esta caché, no será necesario realizar la petición ARP ya que la información se extrae directamente de la tabla.

- Con `arp`:

```
pc2:~# arp -na
? (10.0.0.1) at 0A:29:92:55:93:70 [ether] on eth0
```

la opción –n es para evitar la resolución de nombres.

- Con `ip`:

```
pc2:~# ip neighbour show
10.0.0.1 dev eth0 lladdr 0A:29:92:55:93:70 nud reachable
```

Comprobar la conectividad entre dos dispositivos

La herramienta `ping` comprueba si se puede alcanzar una máquina y muestra el tiempo de ida y vuelta (RTT). Para ello, envía un datagrama ICMP-EchoRequest cada segundo. Si la máquina destino es alcanzable, contestará con un datagrama ICP Echo-Reply.

```
pc2:~# ping 10.0.0.1
PING 10.0.0.1 (10.0.0.1): 56(84) bytes of data
64 bytes from 10.0.0.1: icmp_seq=0 ttl=64 time=1.896 ms
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=2.110 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=2.125 ms
^C
--- 10.0.0.1 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss, time 2025ms
rtt min/avg/max/mdev = 1.896/2.044/2.125/0.105 ms
```

Tiene múltiples opciones que se pueden consultar mediante el comando `man ping`. Las más comunes son:

- `-c <número de paquetes>` fija el número de ICMP-Request que se envían
- `-s <tamaño de paquetes>` fija el tamaño de los datos en el paquete ICMP-Request
- `-t <TTL>` fija el valor del campo TTL en la cabecera del datagrama IP.

Si no se fija el número de paquetes a enviar, el `ping` continúa indefinidamente. Para parar su ejecución se usa `Ctrl+C`.

Comprobar el camino hasta el destino

La herramienta `traceroute` envía paquetes UDP (a los puertos entre 33435 y 33449) variando el TTL del datagrama IP. Comienza enviando 3 paquetes UDP con TTL = 1 cuando obtiene la respuesta del primer router al que dichos paquetes llegan (ICMP-Time Exceeded) envía 3 nuevos paquetes pero esta vez con TTL = 2. De esta forma va descubriendo los routers que existen entre dos dispositivos. Cuando finalmente recibe un ICMP-UDPPortUnreachable (se supone que no hay ninguna aplicación escuchando en los puertos destino), sabe que ha llegado a la red destino. Cada vez que se recibe una respuesta, se imprime la información del nodo que envió dicha respuesta y el RTT.

```
pc4:~# traceroute 10.0.0.1
traceroute to 10.0.0.1 (10.0.0.1), 64 hops max, 40
  byte packets
1 14.0.0.1 (14.0.0.1) 2.3 ms 3.3 ms 1.8 ms
2 13.0.0.1 (13.0.0.1) 4.7 ms 5.6 ms 4.8 ms
3 12.0.0.1 (12.0.0.1) 6.3 ms 8.3 ms 7.6 ms
4 11.0.0.1 (11.0.0.1) 8.9 ms 10.5 ms 9.8 ms
5 10.0.0.1 (10.0.0.1) 11.3 ms 10.3 ms 11.7 ms
```

la opción –n se puede emplear para evitar la resolución de nombres.

Captura y análisis de tráfico en la red

Para poder identificar cualquier posible defecto en la configuración o el funcionamiento de una red, la captura y el análisis del tráfico que circula por la red es fundamental. Para la primera parte, la captura, Linux cuenta con la herramienta `tcpdump`. La herramienta permite ir viendo el tráfico mientras se captura o guardar la información en ficheros para su posterior análisis.

`tcpdump` tiene numerosas opciones (vease `man tcpdump`). A continuación se presentan aquellas más relevantes para el desarrollo de las prácticas:

- `-i dev` Interfaz en la que se quiere capturar el tráfico
- `-n` Evita la resolución de nombres
- `-w <fichero>` Fichero donde se guardarán los paquetes capturados en lugar de que estos aparezcan en la consola.
- `-s <tamaño>` Número de bytes de cada paquete que se capturan. Por defecto son 68. Con –s 0 se captura el paquete completo.

`tcpdump` continúa ejecutándose hasta que lo paramos con Ctrl+C.

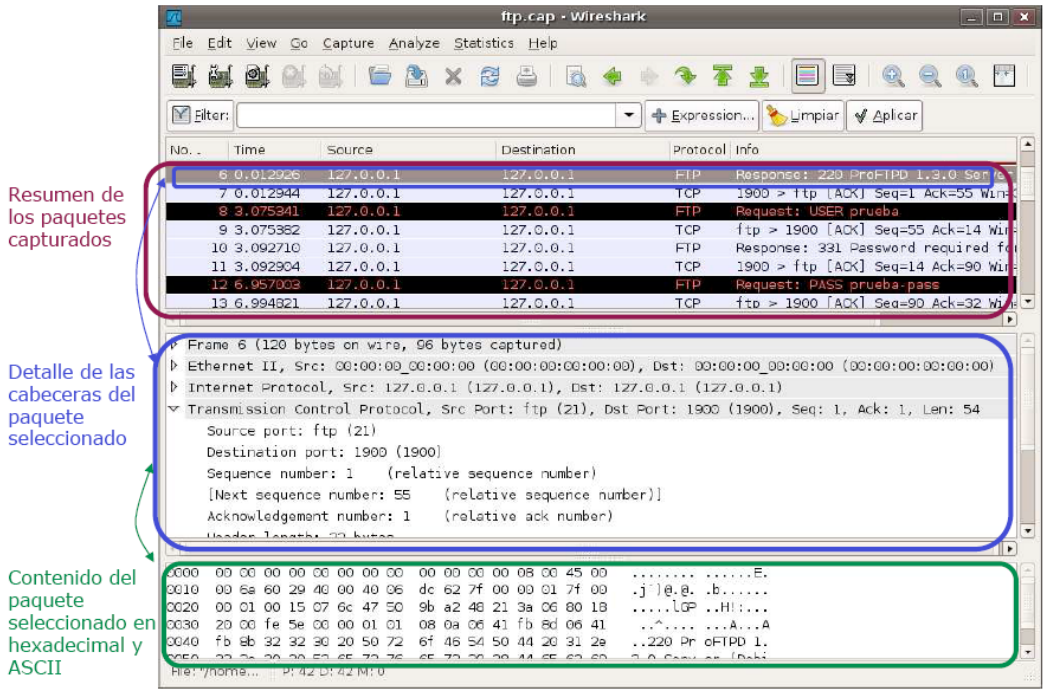
Para el análisis del tráfico capturado, también es posible utilizar tcpdump, pero es más conveniente el uso de analizadores de protocolos gráficos que facilitan el acceso a la información. En el laboratorio emplearemos Wireshark [3]. En el laboratorio Wireshark está instalado en las máquinas reales por lo que es necesario que al hacer la captura mediante `tcpdump`, los ficheros en los que guardemos esas capturas, los pasemos a la máquina real. Para ello, las máquinas virtuales con las que trabajaremos en NetGUI tienen un directorio que se mapea directamente sobre el directorio de usuario de la máquina real. Este directorio es el `/hosthome`. Por lo tanto, cuando se haga una captura es conveniente guardarla en este directorio. Por ejemplo:

```
pc1:~# tcpdump –i eth0 –w /hosthome/pc1.eth0.cap
```

con este comando estamos capturando el tráfico en el interfaz `eth0` de `pc1` y guardando los paquetes capturados en el fichero `pc1.eth0.cap` dentro del directorio `/hosthome`.

Wireshark es una herramienta gráfica que permite visualizar paquetes capturados, navegando a través de los campos de la cabecera y datos de cada uno de los protocolos utilizados.

La siguiente figura muestra la pantalla principal de Wireshark



A través del menú *File->Open*, se pueden abrir los ficheros con las capturas realizadas.

Una característica interesante de Wireshark es que es posible visualizar únicamente los paquetes que sean de un determinado tipo (p.ej. sólo TCP, con destino al puerto 80, etc.). Esto es especialmente útil debido a que durante la captura, en la red se intercambian otros tráficos que aún no siendo de nuestro interés, quedarán almacenados en los ficheros de captura dificultando el análisis ya que tendremos paquetes no deseados entremezclados con los que realmente nos interesan.

Para hacer esto, en Wireshark se pueden definir filtros. En la caja de diálogo dedicada al manejo de estos filtros se puede añadir una expresión que caracterice al tráfico que queremos visualizar. La definición de los filtros tiene una estructura predeterminada que es necesario respetar. Para los usuarios no avanzados, a través del botón *+Expression…*, se abre una ventana que permite editar fácilmente el filtro que deseamos.

A modo de ejemplo se presentan una serie de filtros y su significado:

```
ip.src == 10.0.0.1
```
- Paquetes cuya dirección IP origen sea 10.0.0.1

```
ip.dst == 10.0.0.2
```
- Paquetes cuya dirección IP destino sea 10.0.0.2

```
ip.addr == 10.0.0.3
```
- Paquetes cuya dirección origen o destino sea 10.0.0.1

```
tcp
```
- Todos los segmentos TCP

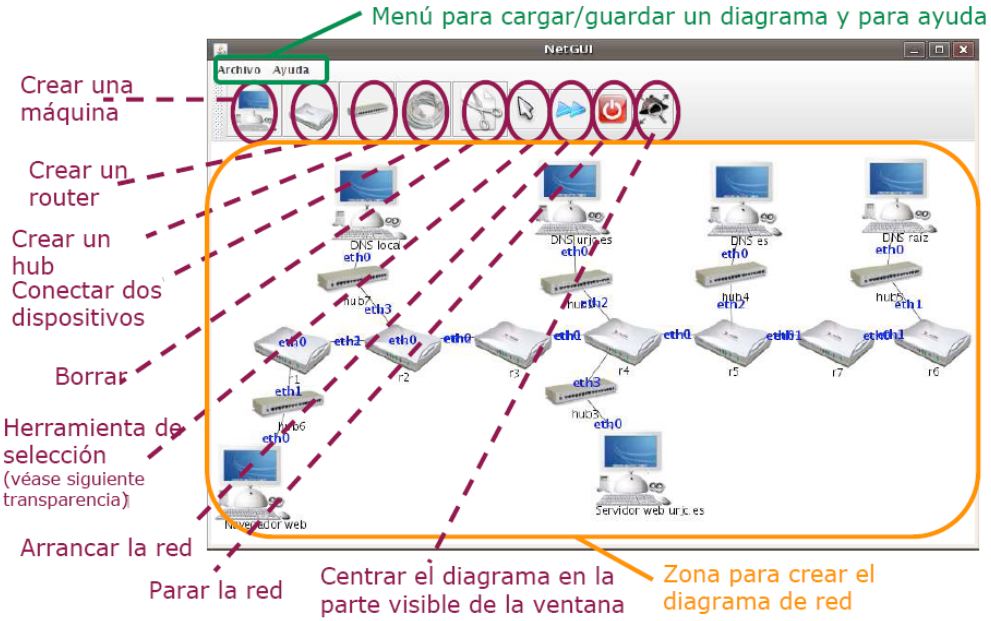
```
tcp.srcport==80 or tcp.dstport==80
```
- Todos los segmentos TCP cuyo puerto origen o puerto destino sea el puerto 80 (tráfico HTTP).

```
ip.addr==10.0.0.2 and tcp.flags.fin
```
- Segmentos TCP con el flag FIN activo y que lleve dirección IP origen o destino 10.0.0.2

Anexo 3: NetGUI

NetGUI se arranca con la orden `netgui`.

La siguiente figura muestra la pantalla principal de NetGUI.



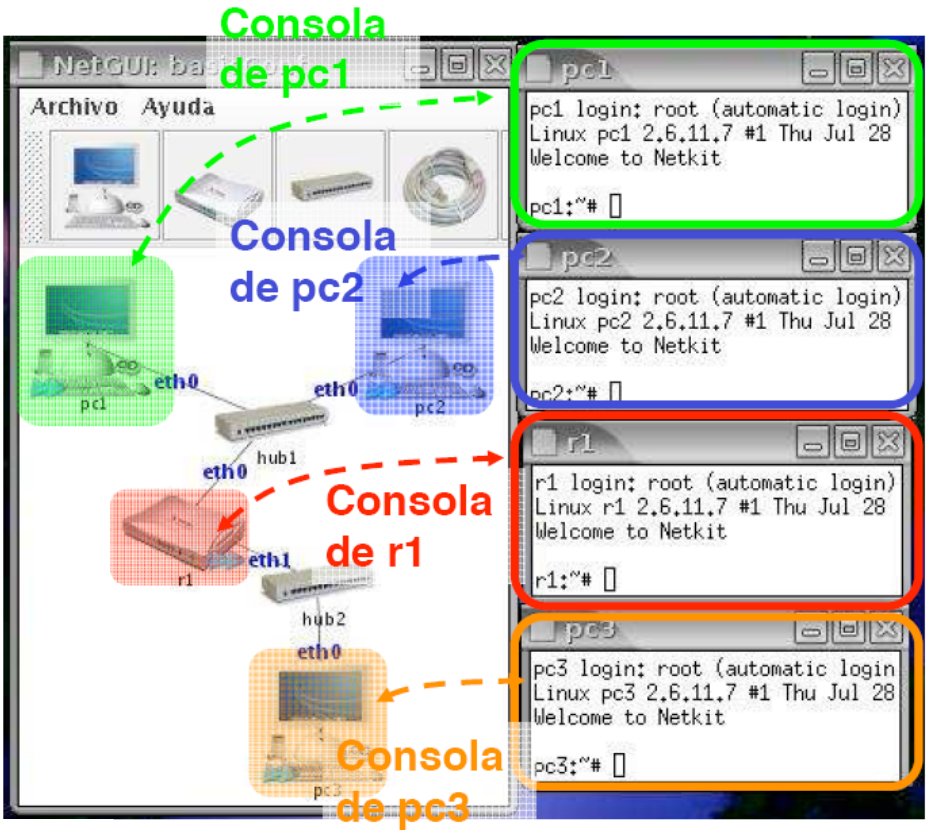
Para empezar a trabajar con NetGUI es necesario visualizar la red con la que queremos trabajar. En NetGUI hay dos posibilidades para ello. La primera es utilizar la zona para crear el diagrama de red y la herramienta de selección para ir creando la red. La segunda es abrir a través del menú *Archivo->Abrir* una red previamente guardada.

La *herramienta de selección* (botón con una flecha con el cursor de un ratón) permite las siguientes funcionalidades:

- **Mover un elemento:** con el botón izquierdo del ratón se pulsa sobre el elemento a mover y se arrastra al destino.
- **Arrancar/Parar un nodo (ordenador o router):** se pulsa con el botón derecho sobre el nodo. Si está parado se arranca, y si esta arrancado se para. Cuando un nodo está arrancado aparecen dos flechas azules en su icono.
- **Mostrar la consola de un nodo arrancado:** cuando se arranca un nodo, se lanza una ventana con la consola de ese nodo. Sobre esa consola es sobre la que se trabajará para la configuración del nodo y el análisis del tráfico que lo atraviesa. Haciendo doble click con el botón izquierdo sobre el nodo aparece en primer plano la ventana de la consola del nodo.
- **Zoom:** pulsando el botón derecho del ratón sobre el fondo de la ventana en cualquier lugar en la que no haya un elemento y moviendo el ratón a derecho o izquierda, se obtiene el efecto de zoom.
- **Desplazamiento:** pulsando el botón izquierdo del ratón sobre el fondo de la ventana en cualquier lugar en la que no haya un elemento y moviendo el ratón

mientras se mantiene el botón pulsado, se puede cambiar la situación de la red dibujada.

En la siguiente figura se muestran las consolas abiertas para las máquinas de una red:



Como se puede comprobar, cada nodo que está arrancado tiene asociada una consola de Linux.

Referencias

[1] Netkit, http://wiki.netkit.org/

[2] NetGUI, http://netlab.sourceforge.net/

[3] Wireshark, http://www.wireshark.org/