PRÁCTICA 7

Simulación y emulación

1 Objetivos

En esta práctica se pretende introducir las funciones de simuladores y emuladores de red, sus diferencias y la utilización práctica de un emulador de red.

2 Material

- PC con software NetGUI

3 Introducción

A la hora de realizar el diseño de una red nueva, realizar pruebas de configuración sobre una red existente o evolucionar una red existente, muchas veces no se dispone ni del equipamiento hardware real ni del escenario con usuarios reales y su carga de tráfico correspondiente. Por tanto, se hace necesario acudir a herramientas software que permitan "simular" o "emular" un determinado escenario de red para evaluarlo.

Los simuladores, consisten en herramientas software que proveen modelos más o menos realistas de los componentes que se quieran utilizar en una red. Pueden tener modelos que representan routers, switches, PCs, etc. para utilizar en la confección de nuestro escenario de red, habitualmente mediante algún tipo de herramienta gráfica. Podremos poner iconos que representen elementos de la red y unirlos mediante líneas que representen su correspondiente cableado. Sobre ese escenario se pueden definir flujos de tráfico de red, y de esta forma servir también para realizar evaluaciones de cual es el efecto final sobre las aplicaciones. Al basarse en modelos, el grado de simplificación de los mismos puede limitar nuestros resultados. Por ejemplo, un simulador puede tener un elemento "router" que reenvíe en función no de una cabecera IP sino de un direccionamiento de red definido a medida en el simulador. Permiten definir escenarios de red complejos y realizar estudios de evaluación de manera muy sencilla (por ejemplo, evaluar el efecto de la variación de la tasa de pérdida de paquetes sobre las aplicaciones). Algunos ejemplos de este tipo de simuladores son OMNET++, ns-2 u OPNET.

Los emuladores consisten en herramientas software que corren sistemas operativos reales de los propios componentes de una red de forma que su funcionamiento es totalmente realista. En el PC emulador se puede tener corriendo instancias de decenas de dispositivos virtuales, e incluso que alguno de los dispositivos virtuales se pueda conectar a la red física. Esto permitiría por ejemplo al usuario de un PC físico conectarse a una red virtual y experimentar su tráfico el mismo efecto que si estuviera conectado realmente a una red física equivalente. Sin embargo, esto supone un sobrecoste de memoria/CPU necesario al tener que correr instancias de cada dispositivo. Si imaginamos por ejemplo un PC, sería correr una máquina virtual del

propio PC, con su sistema operativo (casi) completo. Por tanto, se restringe habitualmente a escenarios de red más pequeños. En Linux tenemos la suerte de disponer de unas funcionalidades a nivel de kernel llamadas User Mode Linux (UML) que permiten lanzar múltiples instancias de máquinas linux con un coste de memoria/CPU muy inferior a los esquemas clásicos de virtualización. Algunos ejemplos de este timo de emuladores son NetKit, NetGUI, gns3, Gini y ClackRouter.

En esta práctica vamos a utilizar un emulador NetGUI, que puede ser muy útil para, por ejemplo, revisar en casa las prácticas que anteriormente ya realizamos sobre los Racks de comunicaciones con equipamiento real.

4 NetGUI

NetGUI (http://mobiquo.gsyc.es/netgui/) es un emulador de red de código libre basado en librarías de emulación NetKit sobre User Mode Linux. De manera totalmente gráfica permite definir escenarios de red, con componentes PCs, routers y switches todos ellos implementados como máquinas virtuales linux. La herramienta nos proveerá con un terminal Linux para la configuración de cada dispositivo.

El emulador NetGUI se ejecuta en un PC linux del laboratorio con el comando:

```
# netgui.sh
```

El emulador nada mas arrancar ofrece una lista de iconos con los elementos que podemos utilizar sobre la zona dedicada para crear el diagrama de red. Basta con pulsar el icono y luego hacer click sobre el escenario para colocar un elemento. Se pueden conectar los equipos con el icono “Conectar dos dispositivos” y haciendo click consecutivamente en ambos dispositivos. En el enlace aparece etiquetado el nombre del interfaz que corresponde al equipo.

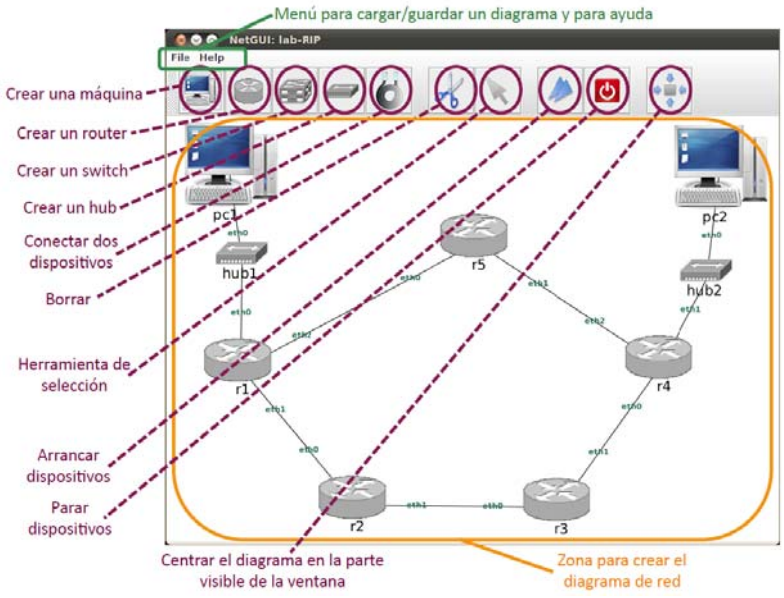

Figura 1.- Ventana de trabajo NetGUI

La herramienta de selección permite la siguiente funcionalidad:

- Seleccionar un elemento: haciendo clic con el botón izquierdo del ratón se selecciona un elemento del escenario de red.
- Mover un elemento: arrastrando con el botón izquierdo del ratón se mueve un elemento dentro del escenario de red.
- Arrancar/Parar un dispositivo (máquina, router o switch): haciendo clic con el botón derecho sobre un dispositivo si está parado se arranca, y si está arrancado se para. IMPORTANTE: Hay que esperar unos segundos para que el dispositivo arranque o se detenga completamente. Cuando un nodo está arrancado aparecen dos flechas azules sobre su icono.
- Mostrar la consola de un nodo arrancado (no aplica al hub): haciendo un doble clic con el botón izquierdo del ratón sobre un dispositivo, su ventana de terminal pasa a primer plano.

En los terminales, serán aplicables todos los comandos que conocemos para la configuración en Linux. Fíjese que el terminal es de root por lo que tendrá permisos de administrador para poder realizar lo que desee. Al arrancar la consola de los nodos el aspecto es el siguiente:

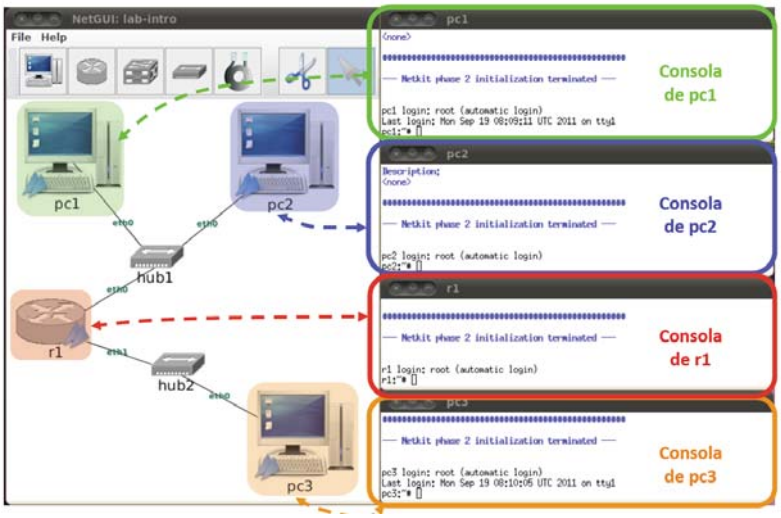

Figura 2.- NetGUI con los terminales lanzados (máquinas virtuales corriendo)

El menú File permite guardar escenarios de red y cargar escenarios guardados previamente. A la hora de guardar con Save, la primera vez hay que elegir un nombre de carpeta (IMPORTANTE: no usar espacios en blanco en el nombre). En esa carpeta se almacenarán todos los ficheros asociados al escenario. Es conveniente guardar nada más terminar de dibujar el escenario, antes de arrancar las máquinas virtuales.

Cuando una ejecución de NetGUI ha terminado de forma incorrecta, se hace imprescindible utilizar antes de lanzarlo de nuevo el siguiente comando:

```
# clean-netgui.sh
```

Para cerrar el emulador:

- NUNCA debe cerrarse NetGUI sin apagar ANTES todas las máquinas virtuales.

- NUNCA debe cerrarse la ventana de una máquina virtual pulsando la X del marco de la ventana. Si se realiza esta acción, el sistema de ficheros de la máquina virtual quedaría incosistente, y aparecerían errores al reiniciar la máquina.

- Para apagar una máquina virtual debe usarse el botón rojo de la interfaz. Si al hacerlo la máquina virtual no se apagase, puede escribirse en su terminal la orden halt y esperar a que la ventana se cierre sola.

- NUNCA debe cerrarse NetGUI pulsando la X del marco de la ventana principal del escenario. Si se hiciera, ya no se podrán apagar las máquinas virtuales a través de NetGUI y habría que hacerlo escribiendo halt en sus ventanas de terminal.

Por lo tanto, el procedimiento adecuado para salir de NetGUI es:

1. Apagar una a una las máquinas virtuales mediante la interfaz de NetGUI.

2. Si alguna máquina virtual no pudiera apagarse mediante la interfaz, apagarla escribiendo halt en su ventana de terminal

3. Si ha habido cambios en el dibujo del escenario que se quieran guardar, elegir en el menú File -> Save.

4. Elegir en el menú File -> Exit.

Pueden pasarse ficheros de las máquinas virtuales a la máquina real que corre el emulador:

- Dentro de una máquina virtual de NetGUI, escribir en el directorio /hosthome permite guardar ficheros en la máquina real: todos los ficheros grabados en el directorio /hosthome estarán en realidad en el $HOME del usuario en la máquina real.

- Las capturas realizadas con tcpdump en las máquinas virtuales conviene guardarlas en /hosthome para que sean accesibles desde la máquina real y poder abrirlas con wireshark por ejemplo.

5 Escenario bifurcación switch-router

5.1 Confeccione el siguiente escenario en NetGUI, donde:

pc1: dirección IP 10.1.1.1/255.255.255.0
router por defecto: el que corresponda
pc2: dirección IP 10.1.1.129/255.255.255.0
router por defecto: el que corresponda
router1: dirección IP interfaz con hub1: 10.1.1.254/255.255.255.0
dirección IP interfaz con hub2: 10.1.2.254/255.255.255.0

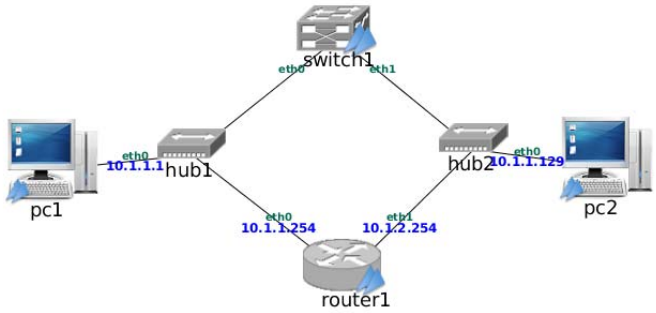

Figura 3.- Escenario bifurcación switch-router

5.2 Verifique si router1 tiene activado el enrutamiento entre sus interfaces.

5.3 Verifique cómo el switch1 hace funciones de conmutación entre las diferentes interfaces ethX de la máquina linux. Para ello existe el comando "`brctl`" que permite configurar la conmutación entre puertos. Con "`brctl show`" puede listar los interfaces entre los que está establecido un switch y el nombre del puente.

5.4 Con "`brcrl showmacs <nombredelpuente>`" muestra la tabla de MACs aprendidas por el switch (también incluye las propias MACs del switch). Haga un ping de pc1 a una IP de su subred que no exista y observe como aparece en esta tabla. Observe cuando se resetea el contador de edad de esa entrada en la tabla parando y arrancando el ping. Si hace el ping de pc1 a pc2 ¿Qué direcciones MAC habrá asociado a sus puertos el conmutador?

5.5 Haga un ping de pc1 a pc2 ¿Por donde van los paquetes a pc2, vía switch1 o router1? ¿Por qué? ¿Cómo lo puede comprobar en el escenario?

5.6 Suponiendo que estamos en pc1 y queremos asegurarnos por donde va el paquete de ping sin tener acceso al resto de nodos de la red ¿Cómo podría saber por donde está mandando los paquetes pc1 solamente mediante comandos ejecutados en el terminal de pc1? Identifique todas las formas que sea posible.

5.7 Determinar el menor número de cambios de configuración que permita hacer que los paquetes sigan el otro camino al seguido en el apartado 5.4

Punto de control 7.1: Avise al profesor cuando haya completado las prácticas hasta este punto.

6 Escenario IP aliasing y control de tráfico

6.1 IP aliasing es el nombre que recibe la asignación de varias direcciones IP de manera simultánea a un interfaz de red. En ciertas ocasiones es muy útil por ejemplo para que una máquina pueda estar conectada simultáneamente a dos subredes IP diferentes con una única tarjeta de red, o que un router sea capaz de enrutar entre diferentes subredes disponiendo solamente de un interfaz de red. En linux, se pueden configurar varias direcciones IP a un interfaz de la siguiente forma:

```
# ifconfig ethX:Y <ip> netmask <netmask>
```

Donde X es el número de interfaz físico (0 para eth0) e Y es el número de interfaz virtual, de los que se pueden definir tantos como se necesiten comenzando por 0. De esta forma al listar los interfaces con ifconfig –a aparecerán interfaces virtuales eth0:0 eth0:1, etc. El manejo de los interfaces virtuales es equivalente al de los interfaces físicos a todos los efectos.

6.2 Confeccione el siguiente escenario en NetGUI, donde:

pc1: dirección IP 10.1.1.1/25
router por defecto: el que corresponda
pc2: dirección IP 10.1.1.129/25
router por defecto: el que corresponda
router1: direcciones IP interfaz con hub1: 10.1.1.126/25 y 10.1.1.254/25. Asigne la primera IP a la interfaz real y la segunda IP a una interfaz virtual.

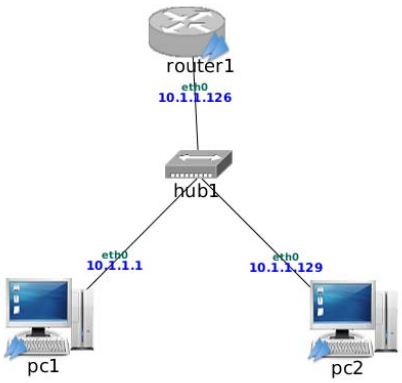

Figura 4.- Escenario IP aliasing y control de tráfico

6.3 Verifique la conectividad entre pc1 y pc2. Mediante el comando ping determine el tiempo RTT entre ambas máquinas, y mediante el comando traceroute verifique que los paquetes entre pc1 y pc2 (en ambos sentidos) circulan vía el router.

6.4 Los interfaces virtuales de router1 comparten la misma dirección física ¿Supone esto algún problema para la conectividad?

6.5 Un router con linux nos permite aplicar técnicas de control de tráfico para por ejemplo introducir retardo extra en los paquetes o introducir pérdidas de paquetes artificiales. Para ello tenemos las herramientas de configuración tc o iptables;

usaremos la segunda.

Para que el router introduzca unas pérdidas del 10% en los paquetes que reenvía:

```
# iptables -A FORWARD -m statistic --mode random --probability 0.10 -j
DROP
```

6.6 Verifique que el router pierde paquetes aproximadamente a la tasa establecida, mandando paquetes ICMP Echo Request uno cada 10ms desde pc1 a pc2. En el manual de ping existe una opción para configurar el tiempo entre paquetes de petición

6.7 La herramienta Netcat permite implementar funcionalidades de cliente/servidor UDP/TCP. Coloque un servidor TCP en el puerto 99 de pc2 con:

```
pc2# nc -l -p 99
```

Y lance un cliente TCP en pc1 que se conecte al servidor anterior

```
pc1# nc 10.1.1.129 99
```

Observe que lo que teclee en un extremo se manda al otro extremo y se vuelva en pantalla.

6.8 Vamos a desactivar los mecanismos de Fast retransmit en ambos extremos, con el comando:

```
# echo 0 > /proc/sys/net/ipv4/tcp_fack
```

6.9 Capture mediante tcpdump el tráfico de la conexión TCP anterior (identifique donde y cómo correr el tcpdump). Se recomienda correr tcpdump con un filtro que sólo muestre el tráfico TCP y evitar así ver otros paquetes del escenario:

```
# tcpdump -i eth0 tcp
```

6.10 Pare con Ctrl+C el cliente y servidor. Ejecute sólo el cliente en pc1 (con el servidor parado) y verifique qué paquetes TCP circulan ¿por qué?

6.11 Capture el tráfico con tcpdump con la opción –w permite guardar la captura en fichero, en formato pcap. Si guarda el fichero en /hosthome podrá luego abrirlo con wireshark de su máquina real. Ejecute ahora el servidor en pc2, y conéctese con el cliente. Intente escribir varias líneas de pocos caracteres y observe como en determinados momentos lo que se escribe no se refleja de manera inmediata en el otro extremo ¿Por qué?

6.12 Analice el fichero de captura anterior y revise en especial:

- Paquetes de establecimiento y cierre de la conexión.
- Paquetes duplicados ¿existen, por qué?
- Números de secuencia.
- Ventana anunciada por el receptor.
- Pérdidas y retransmisiones.

- Datos TCP transportados en cada paquete.

- En el wireshark, en el menú Statistic>Flow Graph… seleccione TCP flow y OK, para ver el diagrama de mensajes.

6.13 Repita la captura ahora con una conexión TCP en la que vamos a mandar muchos datos. Para generar los datos usamos el comando `yes` que vuelva en pantalla caracteres "y" a toda velocidad. Pasamos esos datos como entrada al cliente, y para evitar verlos en pantalla del servidor, redireccionamos su salida a /dev/null (básicamente descartar esa información). En concreto, los comandos en cada extremo serán:

```
pc1# yes | nc 10.1.1.129 99
pc2# nc -l -p 99 > /dev/null
```

Analice la captura e identifique:

- El tamaño de los paquetes ¿Por qué estos tamaños?

- Examine el contenido de datos de los paquetes.

- En el menú de wireshark Statistics>TCP Stream Graph>Time-sequence Graph (Stevens), puede visualizar la evolución del número de secuencia con el tiempo. Razone el perfil de la gráfica.

Punto de control 7.2: Avise al profesor cuando haya completado las prácticas hasta este punto.