Práctica 3

Objetivos de la práctica

- Configuración de red en sistemas Linux y Windows (Tarea 1).
- Automatización del tratamiento de direcciones IP (Tarea 2).

Conocimientos previos:

- □ Redes: rangos y máscaras.
- □ Enrutamiento en IP y tablas de enrutamiento.

Tarea 1: Comandos de información y configuración de red en Windows y MAC/Linux

Tome capturas de pantallas de los comandos y sus resultados en todos los ejercicios.

Paso 1: Obtención de información de la red en su sistema anfitrión

Al tener muchos interfaces (muchos de ellos ficticios usados por los sistemas de virtualización) es complicado saber cuál es realmente el correspondiente a la tarjeta de red que está en funcionamiento. Una forma de obtener la IP que se usa para los envíos externos es de la siguiente forma:

- Capturar con wireshark un `ping` (mensaje ICMP Echo Request) y ver cuál es la IP de origen (también puede usar una trama de la práctica anterior).

Una vez conocida su IP, realice el siguiente ejercicio:

Ejercicio 1 (versión Windows). El comando `ipconfig /all` de Windows muestra información sobre las interfaces de red de la máquina. Ejecute dicho comando en un terminal y, busque la información de su interfaz física e identifique su IP, máscara y puerta de enlace asociada (haga una captura y márquelas). También apunte el campo denominado **Descripción** (lo usaremos más adelante) ¿Cuál es el identificador de su red?

Ejercicio 1 (versión MAC/linux). El comando `ifconfig` de MAC/Linux (o en `ip` algunos Linux) muestra información sobre las interfaces de red de la máquina. Ejecute en un terminal:

```
MAC:   ifconfig; netstat -rn | grep "UGS" | awk '{print "Pasarela: " $2}'
Linux: ip ad sh; ip ro sh | grep "default" | awk '{print "Pasarela: " $3}'
```

busque la información de su interfaz física asociado a su IP, máscara y puerta de enlace asociada (haga una captura y márquelas). También apunte el nombre de dicho interfaz (la primera palabra delante de la configuración que suele tener alguna de las siguientes formas: ethX, wlpXsY, enpXsY, ensX…) que lo usaremos más adelante (lo denominaremos **Descripción**) ¿Cuál es el identificador de su red?

Paso 2: Información básica de Linux (en una máquina virtual)

Para esta práctica, en el campus virtual se ofrece una máquina virtual para realizar la práctica. Descárguela y descomprímala. Para utilizarla necesita disponer de un sistema de virtualización (VirtualBox o VMWare). Si no dispone de ninguno, se recomienda instalar VirtualBox.

Instalación y configuración de la máquina virtual en **VirtualBox**:

- En el menú `Archivo` pulse `Importar servicio virtualizado…`
- En la primera ventana busque el fichero descargado `lUbuntuRySD.ovf`
- En la siguiente ventana desmarque las casillas `DVD`, `tarjeta de sonido` y `Adaptador de red` y abajo marque `Reinicializar la dirección MAC de todas las tarjetas de red.`
- Dele al botón de `Importar.`
- Una vez importada, configure la interfaz de red en la máquina virtual `lUbuntuRySD` en modo *puente*: `Configuración -> Red -> Conectado a: "Adaptador puente"` y en el campo `nombre` debe aparecer el interfaz **Descripción** que apuntamos en el ejercicio 1.

Instalación y configuración de la máquina virtual en **VMWare**:

- Pulse el botón `Open a Virtual Machine`
- En la primera ventana busque el fichero descargado `lUbuntuRySD.ovf`
- Dele al botón de `Importar.`
- Si le sale alguna advertencia pulse `Retry`.

- Una vez importada, configure la interfaz de red de en máquina virtual lUbuntuRySD en modo *puente*: Edit virtual machine settings -> Network Adapter -> Bridged: Connected… y pulse botón Configure Adapters y solo dejar marcado el interfaz que tiene como **Descripción** para que apuntamos en el ejercicio 1.

Arránquela y entre en el usuario alumno (clave alumno).

Ejercicio 2. Observe los datos de la red en Linux usando el comando ifconfig en un terminal. Aparecerá la configuración de dos interfaces: uno llamado **lo** y otro cuyo nombre puede variar (lo utilizaremos en otros ejercicios y nos referiremos a él como **interfazReal**). Observe en primer lugar el interfaz **lo**, ¿para qué se usa este interfaz? Ahora analice el otro interfaz (**interfazReal**), de acuerdo a esos datos (dirección IP y máscara), ¿está el Linux de la máquina virtual en la misma red IPv4 que el Windows (MAC/Linux) de la máquina huésped? ¿Por qué?

Ahora ejecute los siguientes comandos (reemplace la palabra **interfazReal** por la que encontró en el ejercicio 2):

```
sudo /etc/init.d/network-manager stop
sudo kill -9 `cat /run/dhclient-interfazReal.pid`
sudo ip address flush interfazReal
sudo rm /etc/resolv.conf
```

El primer comando desactiva el servicio de red, el segundo desactiva el cliente de DHCP, el tercero libera la IP asignada actualmente, y finalmente el último comando elimina la configuración del DNS.

Ejercicio 3. Vuelva a ejecutar el comando ifconfig, ¿Qué IP/máscara tiene activada ahora el **interfazReal**? ¿Por qué tiene ese tipo de configuración?

Paso 3: Configuración de la red en Linux

Ejercicio 4. Configure la IP y la máscara de subred en Linux con el siguiente comando:

```
sudo ifconfig interfazReal dirIP netmask máscara[1]
```

donde los valores de **interfazReal**, **dirIP** y **máscara** son los mismos valores que observó en ejercicio 2.

Ejercicio 5. Intente ahora hacer desde Linux un ping[2] a la IP de loopback (127.0.0.1), la IP del Windows (MAC/Linux) de su propia máquina, a la IP de su router/puerta de enlace y a una máquina externa a la red (intente tanto por nombre **informatica.cv.uma.es** como por IP: **150.214.54.249**) ¿Cuáles funcionan y cuáles no?

Ejercicio 6. Observe la tabla de encaminamiento de su máquina virtual Linux con el comando route ¿Cómo explica esta tabla por qué algunos pings de los anteriores funcionan y otros no?

Ejercicio 7. Además de consultar la tabla de encaminamiento, con el comando route podemos modificarla (necesita ser root, use sudo delante del comando), en concreto podemos (**R** = red, **M** = máscara y **G** = gateway):

a) **Añadir entrada (entrega directa):** sudo route add -net **R** netmask **M** dev **interfazReal**
b) **Añadir entrada (entrega indirecta):** sudo route add -net **R** netmask **M** gw **G**
c) **Añadir entrada (por defecto):** sudo route add default gw **G**
d) **Borrar entrada (red destino):** sudo route del -net **R** netmask **M**
e) **Borrar entrada (por defecto):** sudo route del default

Usando esos comandos realice las siguientes acciones:

I. Añada una entrada de encaminamiento por defecto usando el comando c (como valor de gateway use la misma puerta de enlace que en Windows (MAC/Linux)). Vuelva a probar los pings que fallaron en el ejercicio 5 y comente el motivo por el que ahora funcionan algunos que antes no.
II. Finalmente, cree el fichero **/etc/resolv.conf**[3] con la línea **nameserver 8.8.8.8** (DNS gratuito ofrecido por Google). ¿Funcionan ahora todos los pings? ¿Por qué funcionan los que antes fallaban?

[1] También es posible usar la notación prefijo con ifconfig <dispositivo> <dir>/<prefijo>
[2] Haga los pings con la opción –c 1 para que solo envíe un mensaje ICMP.
[3] Use sudo leafpad /etc/resolv.conf

Paso 4: Encaminamiento en su sistema anfitrión

Ejercicio 8. Cuando se envía un mensaje al exterior de su red local se hacen dos consultas a su tabla de encaminamiento:

- Primero se busca la entrada que nos lleva al destino final. Al ser externa, se escogerá la entrada por defecto, que nos indica que debemos enviar a la puerta de enlace (su router).
- Luego buscamos la entrada para llegar a nuestro router (la entrada que nos permite comunicarnos con los equipos de nuestra red) que nos dirá que esta comunicación se puede hacer por entrega directa.

Observe la tabla de encaminamiento de su equipo (comando `route PRINT -4` en Windows, `netstat -rn` en MAC y `ip route show` en Linux). Haga una captura de pantalla donde se vean todas las entradas de la tabla marcando:

a) Entrada que le permite comunicarse con un equipo su propia red física (diferente al suyo).
b) Entrada por defecto.

Tarea 2: Automatización del manejo de direcciones IP.

Ejercicio 9. Desarrolle un programa para el trabajo con direcciones IP en formato habitual (X.X.X.X). El programa recibirá como parámetro (o pedirá al usuario) una dirección IP (puede suponer que es válida) y hará lo siguiente:

a) Indique a que clase (A, B, C, D u otra) pertenece la IP.
b) Luego debe solicitar al usuario la máscara (como prefijo y puede considerar que el valor es correcto sin comprobarlo) y debe mostrar:
    - Identificador de la red
    - Dirección de difusión
    - Número de IPs disponibles para equipos

Ejemplo de utilización:

```
user@computer:~$ java P3 192.168.45.30 21
La IP es de clase C
Red: 192.168.40.0/21
Difusión: 192.168.47.255
Número de IPs para host: 2046
```

Nota sobre la memoria

- Si elabora la memoria en Word, se aconseja utilizar la plantilla proporcionada para la práctica. En cualquier caso, la memoria debe contener toda la información que se pide en la plantilla y seguir su estructura.
- La memoria de esta práctica se entregará en conjunto a la memoria de la 1 y 2.
- Dicha memoria debe constar de una portada donde se indique el conjunto de prácticas que incluye la memoria, así como todos los datos del alumno.
- La memoria de cada práctica debe empezar en una nueva página.
- No es necesario copiar el enunciado completo de la práctica pero sí debe copiarse el enunciado de cada ejercicio antes de indicar su respuesta. Debe utilizarse algún sistema de estilos que permita distinguir lo que es el enunciado de lo que es la respuesta al ejercicio.
- Para **cada ejercicio** que obtenga la información de algún proceso realizado en el ordenador (traza de Wireshark, comando,...) **realice una captura** (con <alt>+<impr pant> sólo capturaremos la ventana activa actual). Además de incluir la captura se deben utilizar las herramientas de dibujo del procesador de texto usado para marcar la parte donde se observa lo que pide el ejercicio. Finalmente en el texto añada una pequeña descripción de la captura.
- El formato de entrega de las prácticas será **PDF**.
- En cuanto a los códigos, se entregarán los **.java** (no es necesario entregar todo el proyecto).

Ficheros en la entrega del Bloque I:

En la tarea del campus virtual debe subir un fichero comprimido (en .**zip** o **tar.gz**) con los siguientes ficheros (**no** cree subdirectorios):

- **Memoria.pdf:** Documento con la solución a los ejercicios
- **Ficheros de las trazas de Wireshark:**
  - Práctica 1: **p1.pcapng**
  - Práctica 2: **p2e1-2.pcapng – p2e3.pcapng – p2e4.pcapng – p2e5.pcapng – p2e6.pcapng.**
- **Códigos:** fuentes Java: **p1e9.java – p3e8.java.**
- **Otros:** ficheros no solicitados pero que considere importantes (imágenes, ficheros explicativos, documentación externa consultada…).

APÉNDICE 1: Comandos Útiles:

Comandos relacionados con elementos de la red que suelen estar instalados por defecto en la mayoría de los equipos. No es un listado exhaustivo, existiendo otros muchos pero que habitualmente requieren ser instalados explícitamente.

Para obtener ayuda de ellos puede usar el **man** (Linux y Mac) o poner la opción ***/?*** (Windows). Los comandos marcados con * en Windows, quiere decir que posiblemente no estén instalados por defecto.

En Linux muchos de los comando mencionados se consideran *deprecated* (se recomienda la familia de comandos **iproute2**) pero siguen siendo utilizados muy frecuentemente y la mayoría de las distribuciones siguen incluyéndolos por defecto (y en las que no, permiten instalarlos con facilidad).

ping (Windows, Linux y Mac):
- Permite el envío de un mensaje ICMP a un equipo remoto para comprobar si está activo y los tiempos.
- Precaución: muchas veces los paquetes ICMP están filtrados por los nodos intermedios y no recibir una respuesta no quiere decir que el equipo remoto no esté activo.
- Mediante el uso de opciones se pueden fijar ciertos parámetros del datagrama IP enviado: tiempo de vida, tamaño, bit de no fragmentar…

tracert (Windows) **traceroute** (Linux y Mac):
- Ofrece un listado de los nodos intermedios por los que pasa.
- Precaución: cuando muestra * indica que el nodo intermedio no está ofreciendo información.

netstat (Windows, Linux y Mac):
- Ofrece información sobre el uso de diversos protocolos de red y conexiones (Tema 4).

arp (Windows, Linux y Mac):
- Permite consultar y modificar la tabla caché ARP (equivalencias entre dir. lógicas y físicas).

ipconfig (Windows) **ifconfig** (Linux y Mac):
- Permite consultar y modificar la configuración básica de la red.

route (Windows, Linux y Mac):
- Permite consultar y modificar la tabla de encaminamiento.

nslookup (Windows, Linux y Mac):
- Permite hacer consultas DNS muy básicas (Tema 5).

dig (Windows*, Linux y Mac):
- Similar al **nslookup** pero permite consultas mucho más detalladas y ofrece más información.

ssh (Windows*, Linux y Mac):
- Permite la conexión remota y cifrada a un equipo remoto. Una vez conectado puede ejecutar comandos en el equipo remoto (Tema 5).

telnet (Windows, Linux y Mac):
- Similar al **ssh** pero las comunicaciones se usan sin cifrar lo que lo hace muy vulnerable.
- Otro uso, es permitir establecer conexiones TCP (Tema 4) a cualquier puerto y utilizarlo para acceder/depurar ciertos servicios.

curl/wget (Windows*, Linux y Mac):
- Permite descargar recursos mediante HTTP (Tema 5) mediante la línea de comando.
- Es muy útil para automatizar el acceso a servicios REST o páginas web.