Seminario 5: Diagnóstico y resolución de fallos en redes

Guía del alumno

Introducción al entorno virtualizado y herramientas de diagnóstico de fallos en red

A continuación, se resumen las principales herramientas de diagnóstico de red de uso común que permiten al administrador de redes identificar y resolver problemas.

- `ping`: es una de las herramientas más conocidas. Usa dos mensajes ICMP de tipo 8 (*Echo Request*) y tipo 0 (*Echo Reply*).
  El funcionamiento del `ping` es el siguiente. El origen envía un mensaje *Echo Request* al destino. Si el destino está disponible, este le envía un *Echo Reply*. Una vez el mensaje vuelva al origen, el comando ping muestra por pantalla el número de secuencia, el campo TTL y el cálculo del RTT. Al finalizar el `ping`, se muestra un resumen con estadísticas de los paquetes transmitidos, los paquetes recibidos correctamente, el porcentaje de paquetes que se han perdido y el RTT.

    - `ping -R`: muestra la ruta de ida (del *Echo Request* hacia el destino) y vuelta (del *Echo Reply* hacia el origen). El RTT que calcula no es por salto, sino el del total de ida y vuelta a diferencia del `traceroute` que lo hace por saltos.
    - `ping -n`: muestra las direcciones IP en lugar de los nombres de dominio.

- `traceroute` (en sistemas operativos como GNU/Linux o Mac, o `tracert` en Windows): este comando muestra salto a salto el flujo de tráfico que hace un paquete UDP (es el paquete que se usa por defecto en `traceroute`) desde un emisor a un receptor, trazando la ruta hasta llegar al destino. De esta forma se puede conocer qué punto de la red está fallando y no deja realizar la conexión entre esos equipos. Cuando se ejecuta este comando se obtienen estadísticas del RTT o la latencia de red. Además, también indica la dirección IP de cada uno de los nodos por los que va pasando el paquete hasta llegar a su destino.
  El funcionamiento de `traceroute` es el siguiente (ver Figura 1). Se comienza enviando un paquete UDP al destino con TTL = 1 (el valor de TTL determina cuántos saltos puede atravesar un paquete antes de que se devuelva al origen un mensaje de tiempo excedido de ICMP) y los siguientes paquetes a enviar incrementan el campo TTL en 1 tras recibir el mensaje ICMP anterior.

    - `traceroute -I`: usa ICMP para las pruebas (igual que `ping`).
    - `traceroute -T`: usa TCP SYN para las pruebas.

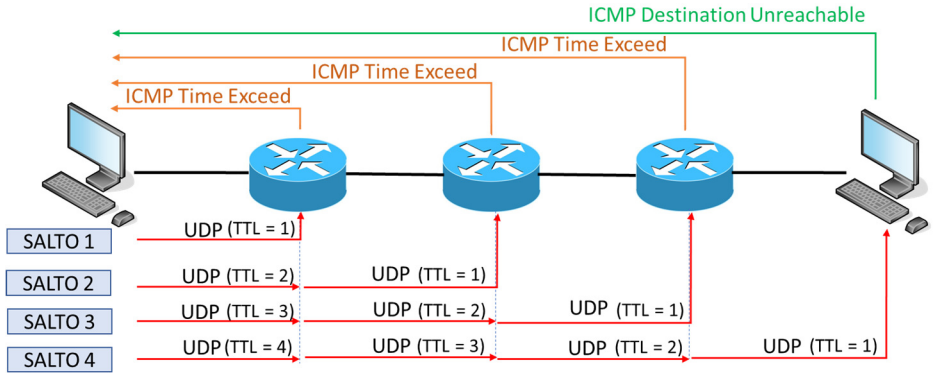

Figura 1. Esquema de funcionamiento de `traceroute`.

- `netstat`: herramienta que muestra todos los puertos y conexiones abiertos en una máquina. Para los puertos de escucha, si la dirección de origen es 0.0.0.0, está escuchando en todas las interfaces disponibles. Si hay una dirección IP en su lugar, entonces el puerto está abierto sólo en esa interfaz específica. Al ejecutar esta herramienta en el terminal, se muestran las direcciones IP de origen y destino, así como los puertos de origen y destino. Los campos Recv-Q y Send-Q muestran el número de bytes pendientes de reconocimiento en cualquier dirección. Finalmente, el campo PID/Nombre del programa muestra el ID del proceso y el nombre del proceso responsable del puerto o conexión de escucha.

    - `netstat -tln`: muestra los puertos que usan TCP en modo *escucha* con el puerto en formato número. La opción `u` (en lugar de `t`) lista los puertos que usan el protocolo UDP.
    - `netstat -tn`: muestra los puertos que usan TCP con conexiones *establecidas* con el puerto en formato número.

  Para mostrar la utilidad de `netstat` se puede utilizar la siguiente herramienta de red:

    - `netcat`: permite abrir puertos TCP/UDP en un host y realizar el rastreo del tráfico en esos puertos. También se puede transferir cualquier tipo de archivo.
      Ejemplo de uso:
      En el servidor:
      ```
      # nc -l 12345
      # netstat -tln
      ```
      En el cliente (misma máquina, distinta terminal):
      ```
      # nc localhost 12345
      # netstat -tn
      ```

- `tcpdump`: es una herramienta de captura de paquetes que se utiliza para solucionar problemas de conectividad de red (muy parecido a `wireshark`), sólo que más liviano y se ejecuta en la línea de comandos.

    - `tcpdump -D`: muestra todas las interfaces disponibles.
    - `tcpdump -n -i [nombre_interfaz]`: captura paquetes IP en esa interfaz y muestra la información (direcciones IP, puertos) en formato numérico.

- `wireshark`: software *open-source* de monitorización y análisis de tráfico de red, que suele usarse como analizador de protocolos. Sirve como una herramienta didáctica para el estudio de las comunicaciones y para la resolución de problemas de red. Se pueden visualizar los campos de cada una de las cabeceras y capas que componen los paquetes monitorizados, proporcionando un gran abanico de posibilidades al administrador de redes a la hora de abordar ciertas tareas de análisis de tráfico.

Red virtual

En la Figura 2 se muestra la red virtual sobre la que trabajaremos en el seminario. Esta red, basada en la topología que hay en el laboratorio, estará montada en el fichero OVA que deberán descargar e importar en VirtualBox los alumnos, llamado `Red_S5_alumnos.ova`.

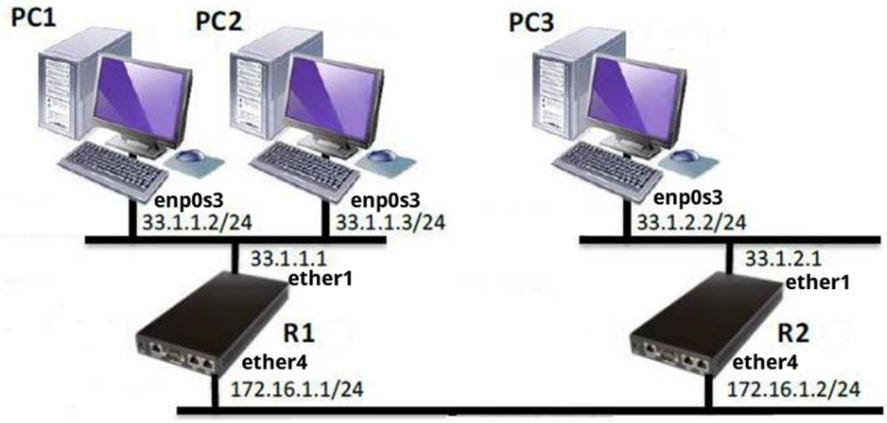

Figura 2. Esquema de red virtualizada.

Ejecución del seminario paso a paso

Una vez tengan instalado el entorno de red virtual (compuesto por 3 PCs y 2 routers), podrán comenzar con el desarrollo del seminario.
En él, se irán detectando distintos problemas de conectividad entre los equipos de la red virtual, y aprenderán a corregirlos.

Fallo 1: Máquinas virtuales no encendidas

Inicie el PC1. Realice un ping a PC2. ¿Funciona?
Compruebe que PC2 está encendido. Si no lo está, enciéndalo, así como el resto de PCs y routers.

Fallo 2: Adaptador no habilitado en VirtualBox en PC1

Cuando estén todos los equipos de la red encendidos, ejecute el comando `ifconfig` para visualizar todas las interfaces de red que existen en los PCs. Compruebe que existan todas las interfaces denominadas `enp0s3` y `lo` (loopback). ¿Falta alguna?

En caso de que falte alguna interfaz, trate de levantarla (habilitarla):

```
# sudo ifconfig enp0s3 up
```

Interprete el error que aparece: "No such device". Para habilitar la tarjeta de red desde VirtualBox apague el equipo (PC1) y en el menú VirtualBox → Configuración → Red habilite el Adaptador 1 (red Interna).

Habilitar la tarjeta de red desde VirtualBox es equivalente a instalar en un PC real (no virtual) una tarjeta de red. Tenga en cuenta que esto es distinto a habilitar/deshabilitar una tarjeta de red desde el sistema operativo. Para hacer esto último, puede usar el siguiente comando:

- Para deshabilitar la interfaz:
  ```
  # sudo ifconfig [nombre_interfaz] down
  ```
- Para habilitar la interfaz:
  ```
  # sudo ifconfig [nombre_interfaz] up
  ```

Este comando realiza los cambios de forma temporal (es decir, los cambios no se guardan al apagar la máquina). Si queremos que los cambios se mantengan al apagar y encender el PC, es necesario modificar la configuración de la interfaz en el archivo `/etc/network/interfaces`. Por ejemplo, para deshabilitar la interfaz `enp0s3`, habría que incluir la siguiente información:

```
auto enp0s3
iface enp0s3 inet manual
      pre-up ifconfig $IFACE down
      pre-down ifconfig $IFACE down
      down ifconfig $IFACE down
```

Fallo 3: Cable no conectado en PC1

Una vez habilitada la tarjeta de red en VirtualBox, ejecute `ifconfig` en PC1. ¿Observa alguna diferencia entre la información que aparece sobre la interfaz `lo` y la interfaz `enp0s3`? Preste atención a las estadísticas (por ejemplo, el número de paquetes transmitidos o recibidos).

Realice un `ping` a PC2 y analice la información que aparece en pantalla. ¿Qué puede estar indicando el mensaje "red inalcanzable"?

Compruebe si hay cable conectado a la tarjeta de red en PC1. Para ello, diríjase al menú VirtualBox → Configuración → Red → Adaptador 1 (red interna) → Avanzadas y compruebe que la opción de cable conectado se encuentra activada.

Fallo 4: Dirección IP mal configurada en PC2

En PC1, con el cable ya conectado, pruebe a realizar un `ping` a PC2. ¿Qué puede significar el error "host de destino inalcanzable"?

Compruebe que la dirección IP de la interfaz `enp0s3` en todos los PCs es correcta. Esto lo puede hacer ejecutando `ifconfig` en una terminal.

Para configurar una dirección IP puede hacerlo ejecutando lo siguiente:

```
# sudo ifconfig enp0s3 33.1.1.3 netmask 255.255.255.0
```

El comando anterior permitirá modificar la dirección IP de forma temporal. Si queremos que los cambios se mantengan al apagar y encender el PC, es necesario modificar la configuración de red en el archivo `/etc/netplan/01-network-manager-all.yaml`. En PC2, abra este archivo y edite la dirección IP para que sea correcta. Posteriormente, actualice la configuración de red en el equipo ejecutando:

```
# sudo netplan apply
```

Compruebe que puede realizar `ping` con éxito entre PC1 y PC2.

Fallo 5: *Default gateway* no configurado en PC1

Realice un `ping` desde PC1 a PC3 y analice el error que se muestra por pantalla. ¿Qué puede indicar "red inalcanzable"? Verifique con `traceroute` si los paquetes transmitidos salen de PC1.

A continuación, realice el `ping` desde PC3 a PC1. ¿Por qué ahora el error es "red de destino inalcanzable"? Verifique con `traceroute` si los paquetes transmitidos salen de PC3 y alcanzan el destino.

Para modificar la puerta de enlace predeterminada o *default gateway* se puede realizar ejecutando el siguiente comando:

```
# sudo route add default gw 33.1.1.1
```

Puede comprobar si esta información se ha añadido como entrada en la tabla de encaminamiento del PC mediante el siguiente comando:

```
# route -n
```

Con el anterior comando es posible que la ruta introducida se almacene sólo de forma temporal. Para que este cambio sea permanente, debe modificar la configuración de red en el archivo `/etc/netplan/01-network-manager-all.yaml`. En PC1, abra este archivo y edite la línea `gateway4` para que la configuración sea correcta. Es importante que dicha línea quede bien alineada a la anterior, de lo contrario dará error. Posteriormente, actualice la configuración de red en el equipo ejecutando `netplan`.

Tras realizar el cambio, compruebe si el `ping` desde PC1 a PC3 funciona. En caso negativo, realice también el `ping` al revés, desde PC3 a PC1, y observe las diferencias en los errores mostrados en pantalla. ¿Por qué desde el PC1 el `ping` se queda "colgado", mientras que desde PC3 el error es "red de destino inalcanzable"?

Fallo 6: Tabla de encaminamiento incompleta en R2

Compruebe con `traceroute` desde PC1 a PC3 si se alcanza el router R2. Analice mediante `wireshark` ejecutándose en PC3 si se capturan los paquetes ICMP de un `ping` desde PC1 a PC3. Para lanzar `wireshark`, ejecute:

```
# sudo wireshark-gtk
```

Realice también un `ping` desde PC3 a PC1 y analice los paquetes capturados con `wireshark` ejecutándose en PC3. Trate de explicar lo que está ocurriendo.

Entre a R2 con *Winbox* desde el PC3. Para ello, desde una terminal diríjase al directorio `Desktop/Software` y ejecute:

```
# wine winbox.exe
```

En el programa, escriba la dirección IP del router (33.1.2.1) y rellene las credenciales de acceso (usuario `admin`, sin contraseña). A continuación, seleccione en el menú IP → Routes y observe la tabla de encaminamiento del router. ¿Falta alguna entrada que sea necesaria para que haya conectividad entre PC1/PC2 y PC3?

Añada la ruta que falta para encaminar tráfico hacia la subred de PC1/PC2. Tenga en cuenta que la red de destino es 33.1.1.0/24 y la dirección IP de la puerta de enlace es 172.16.1.1.

Compruebe que el `ping` entre PC1/PC2 y PC3 funciona.

Fallo 7: Tráfico de TELNET restringido por el firewall de R2

Pruebe a realizar `telnet` desde PC1 y PC2 a PC3. Para ello, ejecute:

```
# telnet 33.1.2.2
```

¿Por qué se queda la terminal "colgada" sin establecer la conexión? Use `wireshark` en PC3 para verificar si la petición de conexión llega a PC3. Recuerde ejecutar esta herramienta con el comando `sudo wireshark-gtk`.

Acceda a R2 con *Winbox*. A continuación, seleccione en el menú IP → Firewall y observe el contenido de la tabla. Las reglas que aparecen son las siguientes:

- #0: chain: forward, dst. address: 33.1.2.0/24, protocol: icmp, action: accept
- #1: chain: forward, src. address: 33.1.2.0/24, protocol: icmp, action: accept
- #2: chain: forward, action: drop
- #3: chain: forward, dst. address: 33.1.2.0/24, protocol: tcp, dst. port: 23, action: accept
- #4: chain: forward, src. address: 33.1.2.0/24, protocol: tcp, src. port:23, action: accept

El significado de estas reglas es el siguiente:

- #0: permite el tráfico tipo ICMP (`ping`) con destino cualquier IP de la subred 2 (PC3)
- #1: permite el tráfico tipo ICMP (`ping`) procedente de cualquier IP de la subred 2 (PC3)
- #2: descarta todo el tráfico que atraviese el router
- #3: permite el tráfico `telnet` (TCP, puerto 23) con destino la subred 2 (PC3)
- #4: permite el tráfico `telnet` (TCP, puerto 23) procedente de la subred 2 (PC3)

Tenga en cuenta que las reglas se comprueban de arriba abajo y sólo se "dispara" la primera que se cumpla. ¿Es necesario realizar algún cambio en las reglas para que el servicio TELNET funcione correctamente?

Mueva la regla `drop` abajo del todo arrastrándola con el ratón. Esta regla no deja pasar ningún tráfico a través del router. Se suele poner la última, de modo que todo el tráfico que no se haya dejado pasar con las reglas anteriores, se descartará.

Pruebe a realizar `telnet` desde PC1 y PC2 a PC3. ¿Funciona?

Fallo 8: Puerto 23 cerrado en firewall de PC3

Ejecute `netstat -tln` en PC3 para comprobar que hay un proceso (TELNET) escuchando en el puerto 23.

A continuación, use `wireshark` para comprobar si la petición de conexión de TELNET llega a PC3. Filtre los paquetes añadiendo el siguiente filtro en la barra superior: `ip.addr ==`

`33.1.1.2`. Analice los mensajes que aparecen en pantalla. ¿Por qué aparecen sucesivas retransmisiones? ¿Se llega a establecer la conexión TCP al puerto 23?

Compruebe el estado del *firewall* de PC3. Para ello, ejecute:

```
# sudo ufw status
```

Las siglas UFW significan *Uncomplicated Firewall* y hacen referencia a una aplicación que tiene como objetivo establecer reglas en `iptables`, las tablas de *firewall* nativas en los sistemas Linux. Puesto que `iptables` tiene una sintaxis relativamente compleja, utilizar UFW para realizar su configuración es una alternativa útil.

Compruebe observando el resultado si existen reglas que limitan el tráfico del servicio TELNET. Para permitir el tráfico hacia el puerto 23, ejecute lo siguiente:

```
# sudo ufw allow 23

# sudo ufw status
```

Compruebe si es posible realizar una conexión TELNET desde PC1 a PC3. ¿Funciona? Ejecute en una terminal `netstat -tn` para comprobar que se ha establecido la conexión TCP.

Fallo 9: Acceso restringido a PC3 en TELNET

Realice un TELNET desde PC2 a PC3 y compruebe si funciona.

Para controlar el acceso a las aplicaciones, existe un mecanismo denominado *TCP Wrapper*, que consiste en una biblioteca que provee un control de acceso simple y administración de *logs* estandarizada para aplicaciones que lo soporten y reciban conexiones de red. Los *TCP Wrappers* son, por tanto, listas de control de acceso (ACL) basadas en *hosts* y utilizadas para filtrar accesos de red a los servicios locales.

Use `wireshark` en PC3 para comprobar si el paquete llega al destino. Añada el siguiente filtro: `ip.addr == 33.1.1.3`. ¿Se establece la conexión TCP? En caso afirmativo, ¿llega la petición de conexión de TELNET? ¿Qué ocurre con la respuesta?

Abra en PC3 el fichero `/etc/hosts.deny` y observe la línea:

```
in.telnetd: ALL EXCEPT 33.1.1.2
```

¿Qué significado tiene esta línea? Para permitir el tráfico desde PC2 añada su dirección IP al final de la línea, separando las direcciones IP con una coma.

Finalmente, compruebe si funciona TELNET desde PC2 a PC3. Tenga en cuenta que, para salir de TELNET, debe ejecutar el comando `exit`. Preste atención al *prompt* de la terminal para identificar la máquina en la que ejecuta los comandos.