

PRÁCTICA 5: FILTRADO DE TRÁFICO EN ROUTERS

Autor: Rogelio Montaña

OBJETIVO Y DESCRIPCIÓN GENERAL.

Existen diversas ocasiones en las que es conveniente o necesario filtrar determinado tráfico en un router. Algunos ejemplos de tales situaciones son los siguientes:

- Un host está infectado por virus y para evitar que ataque o infecte a otros ordenadores se quiere impedir que envíe tráfico. En este caso se deben filtrar los paquetes que tienen como origen esa dirección IP.
- Un host está distribuyendo ilegalmente música, películas o software (normalmente mediante programas peer-to-peer) y se quiere impedir que dicha distribución se lleve a cabo. En este caso se deben filtrar los paquetes que tienen esa dirección IP como origen o destino.
- Un servidor ofrece sus servicios por un puerto 'bien conocido' (por ejemplo el puerto 80 de TCP en el caso de un servidor web). Se supone que a dicho servidor solo deben llegar paquetes dirigidos al puerto 80. En estos casos suele ser buena práctica filtrar cualquier paquete dirigido a ese servidor que no vaya dirigido al puerto 80 de TCP, ya que en el mejor de los casos dicho tráfico es inútil y en el peor puede tratarse de intentos de ataque a ese servidor aprovechando vulnerabilidades accesibles por otros puertos o por otros protocolos (ICMP, UDP, etc.).
- Un servidor tiene restringido su acceso a una serie de clientes externos autorizados que se identifican por una serie de direcciones IP. En este caso se debería filtrar cualquier petición de conexión entrante que no provenga de una de las direcciones IP autorizadas.
- Se quiere impedir el establecimiento de conexiones TCP entrantes para todos los ordenadores de la LAN, excepto para un conjunto reducido de servidores que deben estar abiertos al exterior (y que se supone que estarán especialmente protegidos). En este caso se debe filtrar cualquier intento de conexión entrante que no vaya dirigido a los servidores.
- Se quiere impedir que los usuarios de la LAN hagan uso de 'IP spoofing', es decir de direcciones IP falsas. Para ello se establece un filtro que comprueba que los paquetes recibidos en la interfaz LAN del router pertenecen a la red que está conectada a esa LAN. Análogamente se comprueba que por la interfaz WAN no lleguen paquetes con dirección de origen perteneciente a la LAN. Este filtro es aplicado de forma habitual por la mayoría de los ISPs.

Todos los casos anteriores podrían resolverse mediante la utilización de un cortafuegos. En realidad un cortafuegos es un router al que se le ha dotado de una interfaz especialmente diseñada para la inclusión de filtros.

En esta práctica los alumnos tendrán la oportunidad de probar diversos mecanismos que permiten realizar filtrado de tráfico en los routers, con lo que podrían hacer frente a situaciones como las anteriormente expuestas. En la inmensa mayoría de los casos esos filtros se configuran haciendo uso de lo que se conoce como Listas de Control de Acceso o ACLs (Access Control Lists).

Para el desarrollo de esta práctica se utilizará una maqueta como la que se muestra en la Figura 1, formada por dos LANs, Norte y Sur, cada una formada por un router, dos hosts y un hub que los interconecta. Los dos routers (norte y sur) están conectados entre sí por una línea serie a través de sus interfaces WAN. Los hosts, que denominamos A (HNA/HSA) y B (HNB/HSB) reciben direcciones IP 10.0.x.100 y 10.0.x.200, respectivamente (x=1 para la LAN Norte, x=2 para la LAN Sur). La interfaz LAN del router recibe la dirección 10.0.x.1. Obsérvese que el router y el host A reciben direcciones de la

Arquitectura de Redes y Servicios

mitad inferior de la red (rango 0-127), mientras que el host B recibe una dirección de la mitad superior (rango 128-255) de la red IP asignada a la LAN.

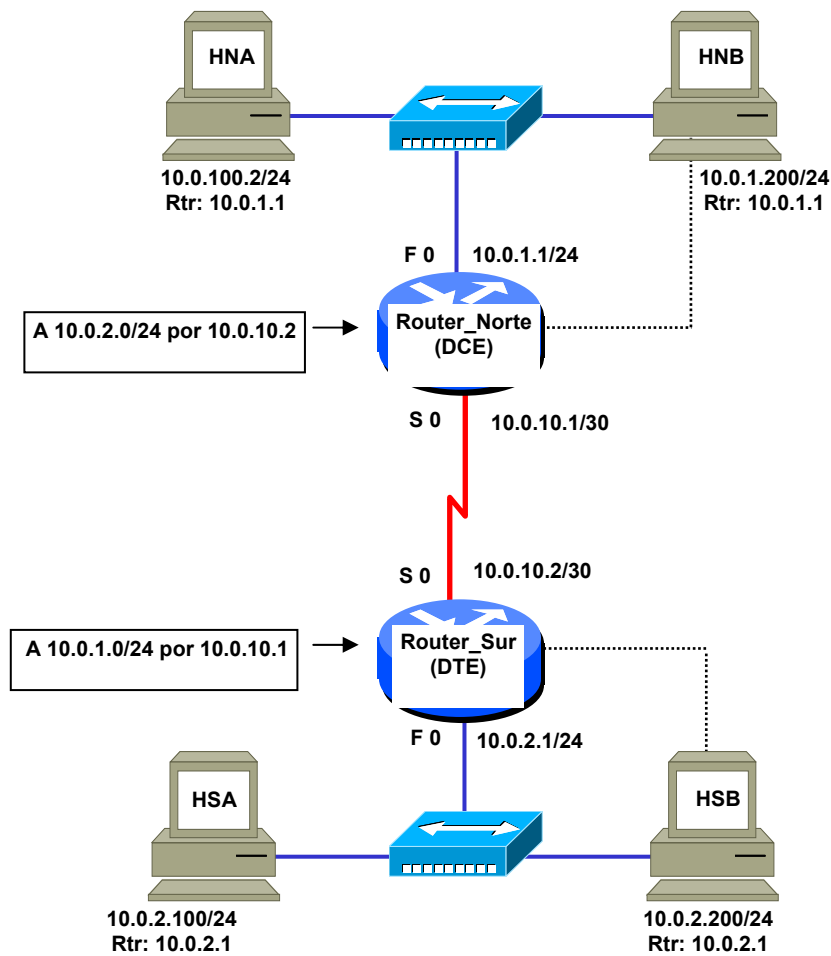


Figura 1. Esquema de la maqueta sobre la que se desarrolla la práctica

Aunque en la Figura 1 se supone que los routers tienen una interfaz WAN y una LAN, y que estas se denominan Serial 0 y FastEthernet 0, respectivamente, las denominaciones exactas pueden cambiar en función del modelo de router utilizado, por ejemplo la interfaz LAN puede ser Ethernet 0 en algún caso. También puede ocurrir que algunos routers tengan más de una interfaz LAN y/o más de una interfaz WAN. En ese caso el profesor indicará a los alumnos cual deben utilizar para el desarrollo de la práctica.

Dado que el objetivo de esta práctica es experimentar con la configuración de filtros en los routers la configuración básica que aparece en la Figura 1 (direcciones de interfaces y rutas) se encuentra ya grabada en la memoria permanente de los routers. Los alumnos solo deberán realizar las modificaciones de la configuración correspondientes a los filtros que vayan a probar, haciendo dichos cambios en la memoria temporal de los routers. De este modo pueden volver en cualquier momento a la situación inicial apagando y encendiendo los routers.

Por lo que respecta a los hosts los alumnos deben proceder a configurarlos con las direcciones y router por defecto que se indican en la figura, haciendo uso de los comandos **ifconfig** y **route** según se indica en el apartado correspondiente del guión.

El número de maquetas utilizadas en cada sesión de prácticas dependerá del número de alumnos. El número ideal es de cuatro a seis alumnos por maqueta. Cada maqueta funciona de forma independiente del resto durante todo el desarrollo de la práctica.

Paso 1. Encendido de los equipos y verificación de las conexiones

En primer lugar los alumnos deberán comprobar que todos los equipos que forman la maqueta están interconectados según se indica en la Figura 1. En caso contrario realizarán las conexiones pertinentes, teniendo en cuenta que todos los latiguillos Ethernet deben ser normales, es decir no cruzados. Para las conexiones de consola se deben utilizar cables planos negros.

Una vez comprobadas las conexiones procedemos a encender los hosts y arrancarlos con el sistema operativo 'linux redes'. Una vez ha arrancado el sistema operativo entraremos con el usuario root y la password que nos indique el profesor y pondremos en marcha el programa minicom mediante el comando '**minicom -s**', pulsando a continuación la tecla escape. Con el programa minicom ya en marcha encenderemos los routers, debiendo ver aparecer por consola los mensajes de arranque. Si no aparece nada deberemos comprobar los cables, los equipos y la configuración del minicom, que debe ser:

- Velocidad 9600 bits/s
- 8 bits de datos
- Un bit de parada (8N1)
- Sin paridad
- Control de flujo: ninguno
- Dispositivo de entrada: /dev/ttyS0

(El uso del dispositivo ttyS0 se debe a que estamos utilizando el puerto COM1 del ordenador.)

Paso 2. Configuración de los hosts.

En primer lugar debemos asignar la dirección IP que corresponde a cada host, según se indica en la Figura 1. Para ello utilizaremos el comando:

```
ifconfig eth0 inet dirección_IP netmask máscara
```

Para comprobar que la asignación se ha efectuado correctamente podemos ejecutar el comando:

```
ifconfig eth0
```

Una vez definida la dirección IP asignaremos a los hosts la ruta por defecto. Para ello utilizaremos el comando:

```
route add default gw dirección_IP
```

poniendo en el campo *dirección_IP* la de la interfaz Ethernet correspondiente al router al que ese host se conecta. Para comprobar que la definición se ha hecho correctamente utilizaremos a continuación el comando:

```
route -n
```

El host debe tener ahora tres rutas definidas que corresponden a la ruta loopback, la ruta de su propia LAN y la ruta por defecto que acabamos de definir. Por ejemplo en el caso de HNA o HNB debe aparecer algo similar a lo siguiente:

```
> route -n
Routing tables
Destination Gateway Genmask Flags Metric Ref Use Iface
10.0.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
127.0.0.1 0.0.0.0 255.0.0.0 U 0 0 0 lo
0.0.0.0 10.0.1.1 0.0.0.0 UG 0 0 0 eth0
```

Las rutas definidas mediante el comando **'route add'** se van añadiendo a la lista existente. Por tanto si nos equivocamos deberemos borrar la ruta incorrecta mediante el comando:

```
route del -net 0.0.0.0 gw dirección_IP
```

Algunos comandos Linux cuando se utilizan direcciones IP intentan realizar la resolución inversa de las direcciones en el DNS, para averiguar el nombre correspondiente. En algunos casos (por ejemplo los comandos **'ping'**, **'route'** o **'traceroute'**) esto puede evitarse con la opción **'-n'**, pero en otros comandos como **'telnet'** no existe esta opción, por lo que es necesario esperar a que expire el timeout del DNS (unos 30 segundos aproximadamente). Para evitar este retardo cuando utilicemos el comando telnet cambiaremos de nombre el fichero **resolv.conf** en el directorio **/etc** mediante el comando:

```
mv /etc/resolv.conf /etc/resolv.conf.old
```

De esta forma evitamos la consulta al DNS y por tanto la espera. Podemos prescindir entonces de la opción **'-n'** en los comandos **'ping'**, **'route'** o **'traceroute'**. Si el fichero **resolv.conf** no existe en el directorio **/etc/** el cambio de nombre dará un error, pero entonces no nos dará problemas así que no debemos preocuparnos.

Ahora desactivaremos en todos los hosts el servicio de cortafuegos mediante el comando:

```
Service iptables stop
```

De esta forma evitamos posibles interferencias entre el cortafuegos del host y los filtros que vamos a definir.

Una vez hemos configurado los hosts comprobaremos mediante el comando ping que existe conectividad entre todos ellos, especialmente con los hosts de la LAN remota.

Paso 3. Filtrado de paquetes por la dirección de destino. Uso de la ruta a Null0

Vamos a suponer en primer lugar que queremos impedir completamente la comunicación de un host concreto con el exterior. Puede ser que ese host esté difundiendo un virus, que esté realizando una distribución ilegal de música, películas, etc.

Una forma sencilla de conseguir ese bloqueo es incluir en el router una ruta host (es decir, una ruta con máscara de 32 bits) que envíe a la interfaz virtual **Null0** los paquetes que lleven como dirección de destino la de ese host. Cuando un router tiene que enviar un paquete por la interfaz **Null0** simplemente lo descarta.

TAREA PARA LOS ALUMNOS

Los alumnos deben configurar ahora en su router una ruta hacia la interfaz Null0 para el ordenador 1 de su LAN. Por ejemplo para el caso del host HSA deberán introducir en **Router_Sur** los siguientes comandos:

```
Router_Sur>ENable
Router#CONFigure Terminal
Router(config)# IP ROute 10.0.2.100 255.255.255.255 Null0
Router_Sur(config-line)#CTRL/Z
Router_Sur#
```

Una vez definida la ruta deberán comprobar mediante el comando ping que no es posible la comunicación desde fuera con el ordenador A.

Pero ¿estamos con esta ruta filtrando realmente todo el tráfico del host A? El ping usado como herramienta de comprobación a veces nos lleva a conclusiones erróneas. Para averiguarlo vamos a realizar ahora la siguiente prueba: arrancaremos el Ethereal en el host B (Norte y Sur) y a continuación lanzamos el ping desde HNA hacia HSB y desde HSA hacia HNB; veremos que el ping sigue sin responder, pero sin embargo los paquetes **ICMP Echo** nos están llegando todos al host B. Solo los mensajes de respuesta **ICMP Echo-reply** están siendo descartados por los routers. Es decir, la ruta hacia **Null0** impide que llegue tráfico al host vetado, pero no impide que éste envíe. Esto se debe a que los routers encaminan los paquetes en base a la dirección de destino, sin analizar la dirección de origen.

Una vez terminada esta prueba los alumnos deben **apagar y encender el router** para forzar una carga de la configuración inicial.

Todas las aplicaciones que se basan en el protocolo TCP, como por ejemplo los programas peer-to-peer, requieren para funcionar la comunicación bidireccional, por lo que un bloqueo en un solo sentido como el que hemos conseguido aquí sería suficiente para paralizar el tráfico en este tipo de situaciones. Pero en el caso de ataques con virus el host infectado puede estar efectuando envíos masivos de tráfico UDP, ICMP o de otro tipo, con lo que puede inundar de tráfico la red sin necesidad de recibir ninguna respuesta. En estos casos un bloqueo unidireccional como el que hemos realizado no es suficiente.

Otra limitación que plantea el uso de rutas hacia **Null0** es la imposibilidad de aplicar el filtrado de tráfico de forma selectiva por interfaz. En nuestro caso esto no es mucho problema puesto que únicamente tenemos dos interfaces, por lo que los paquetes que entran por una necesariamente han de salir por la otra. Pero supongamos que tuviéramos dos líneas serie y que solo quisiéramos aplicar la restricción en una de ellas. No tenemos manera de especificar que la ruta hacia **Null0** solo debe aplicarse al tráfico proveniente de una interfaz.

Paso 4. Filtrado de paquetes por la dirección de origen. Uso de ACLs (Access Control Lists) estándar.

Las listas de acceso, o ACLs, son un mecanismo que permite definir en un router reglas de filtrado que se aplican sobre una interfaz concreta y que hacen referencia a diversos campos de la cabecera IP o de la siguiente cabecera (ICMP, TCP, UDP u otras). Esto nos da máxima flexibilidad en el establecimiento de filtros y nos permite eludir las limitaciones que comentábamos antes respecto al uso de rutas hacia **Null0**.

Las ACLs requieren analizar con mayor detalle la cabecera de los paquetes, motivo por el cual el rendimiento de los routers suele disminuir cuando se aplican ACLs, aunque esta merma es menor en los modelos de gama alta, que disponen de hardware específicamente diseñado para acelerar el proceso asociado con las ACLs.

Existen dos tipos de ACLs, estándar y extendidas. Las ACLs estándar únicamente permiten establecer filtros basados en la dirección IP de origen. Las ACLs extendidas pueden realizar el filtrado en base a prácticamente cualquier campo de la cabecera IP, ICMP, TCP, UDP, etc.

Cada ACL está compuesta por un conjunto de reglas que se evalúan en el orden en que se han declarado. Cuando una regla es aplicable a un paquete (es decir se produce un 'match') se aplica la acción correspondiente (Permit o Deny) y ya no se comprueban el resto de reglas que componen la ACL. Todas las ACLs tienen implícita una regla final que es DENY ANY, de forma que si un paquete no cumple ninguna de las reglas establecidas en la ACL es finalmente descartado.

En un router pueden definirse varias ACL estándar y/o extendidas. Cada ACL recibe un número identificador. Las ACLs estándar se identifican con números del 1 al 99 y las ACLs extendidas con números del 100 al 199.

Una vez definida una ACL se aplica sobre una o varias interfaces, en sentido entrante o saliente. Una misma ACL se puede aplicar sobre varias interfaces, pero sobre una interfaz se pueden aplicar simultáneamente dos ACLs como máximo, una para el tráfico entrante y otra para el tráfico saliente.

Las reglas que componen una ACL estándar se definen en modo Configuración Global mediante el comando '**Access-list**'.

La sintaxis del comando '**Access-list**' para ACLs estándar es la siguiente:

```
Access-list n°_lista Permit|Deny IP_origen [wild-mask]
```

donde '**n°_lista**' es un valor entre 1 y 99 que identifica la ACL. Las distintas ocurrencias del comando '**Access-list**' con un mismo valor del parámetro '**n°_lista**' identifican las diferentes reglas que componen una ACL dada.

Por ejemplo para descartar todos los paquetes cuya dirección de origen pertenezca a la red 147.156.0.0/16 definiríamos la siguiente ACL (identificada en este ejemplo con el número 1):

```
Router#CONFigure Terminal  
Router(config)# Access-list 1 Deny 147.156.0.0 0.0.255.255  
Router(config)# Access-list 1 Permit Any  
Router(config)#CTRL/Z  
Router#
```

Esta ACL está formada por dos reglas: la primera indica que se descarten los paquetes que contengan una dirección IP de origen perteneciente a 147.156.0.0/16, la segunda indica que se deje pasar cualquier paquete. Dado que las ACLs tienen siempre implícita una última regla '**Deny Any**', si se quiere que pueda pasar el tráfico que no ha sido expresamente prohibido es preciso incluir al final la regla '**Permit Any**'. Las reglas se comprueban en el orden en que se han declarado, por lo que si en este caso se hubieran declarado al revés la regla '**Deny 147.156.0.0 0.0.255.255**' nunca se aplicaría puesto que la '**Permit Any**' la satisfacen todos los paquetes..

Obsérvese que en las ACLs el significado de la máscara (aquí llamada 'wild-mask') es inverso al que tiene normalmente, es decir en este caso los ceros indican la parte fija y los unos la parte variable de la dirección.

La definición de la ACL por sí misma no tiene efecto alguno entretanto no se aplica sobre alguna interfaz. Para aplicar una ACL se utiliza el comando '**IP Access-group**' en modo Configuración de Interfaz, sobre la interfaz deseada, en sentido entrante o saliente. La sintaxis del comando '**IP Access-group**' en sentido entrante es:

Práctica 5: Filtrado de tráfico en routers

```
IP ACCEss-group n°_lista In
```

y en sentido saliente:

```
IP ACCEss-group n°_lista Out
```

El sentido (entrante o saliente) se entiende referido siempre al punto de vista del router.

Podemos ver las ACLs definidas mediante el comando '**Show IP ACCEss-lists**' en modo Privilegiado. Además de permitirnos comprobar las reglas que componen cada ACL este comando nos indica cuantas veces se ha satisfecho cada una de las reglas que componen la ACL.

TAREA PARA LOS ALUMNOS

Los alumnos deberán definir una ACL que permita descartar los paquetes cuya dirección de origen coincida con la del ordenador 1 de su LAN. Los paquetes deberán ser descartados lo antes posible en el router.

Una vez configurada y aplicada la ACL comprobarán mediante el comando ping y la herramienta Ethereal en el Host 2 que efectivamente los paquetes ICMP Echo ya no están llegando.

Una vez terminada esta prueba los alumnos deben **apagar y encender el router** para forzar una carga de la configuración inicial.

Para que una ACL deje de aplicarse sobre una interfaz se debe utilizar el comando, en modo Configuración de Interfaz:

```
NO IP ACCEss-group n°_lista In|Out
```

Con esto la ACL deja de aplicarse en esa interfaz, pero sigue definida. Si se quiere borrar la ACL definida se debe utilizar el comando (en modo Configuración):

```
NO ACcess-list n°_lista
```

En una ACL no pueden borrarse, intercalarse ni cambiarse reglas de orden. Si se definen nuevas reglas en una ACL existente se añaden siempre por el final. Si se quiere borrar, intercalar o cambiar reglas de orden se ha de borrar toda la ACL y definirla de nuevo.

Cuando se borra una ACL de un router quedan sin efecto todas las aplicaciones de dicha ACL que hubiera en las interfaces de ese router

Paso 5. Configuración de un filtro anti-spoofing mediante ACLs estándar.

Una aplicación bastante habitual de las ACLs es lo que se conoce como filtro 'anti-spoofing', que consiste en evitar que una red envíe al exterior paquetes cuya dirección de origen no sea suya, y viceversa, que no admita paquetes provenientes del exterior que tengan como dirección de origen alguna suya. Por ejemplo si nuestra LAN fuera la red 147.156.0.0/16 el filtro anti-spoofing se configuraría con la siguiente secuencia de comandos:

```
ACcEss-list 1 Permit 147.156.0.0 0.0.255.255
ACcEss-list 1 DEny Any
ACcEss-list 2 DEny 147.156.0.0 0.0.255.255
ACcEss-list 2 Permit Any
INterface FastEthernet 0
IP ACcEss-group 1 In
INterface Serial 0
IP ACcEss-group 2 In
```

TAREA PARA LOS ALUMNOS

En esta tarea los alumnos deberán configurar en los routers las ACLs necesarias para que se realice el filtrado anti-spoofing de su LAN, suponiendo que la máscara de subred fuera /25, es decir que cada LAN abarcara las direcciones 0 a 127 únicamente. De esta forma los paquetes enviados por el host B (con dirección IP 10.0.X.200) serán tratados como paquetes de IP spoofing. Una vez aplicadas las ACLs deberán comprobar que la comunicación hacia el exterior solo es posible para el host A de la LAN (puesto que el host B, con dirección .200, se encuentran fuera del rango permitido).

Obsérvese que la máscara /25 solo debe afectar a la ACL, no a la configuración de los hosts o de la interfaz del router que debe ser /24 durante toda la práctica.

Una vez terminada esta prueba los alumnos deben **apagar y encender el router** para forzar una carga de la configuración inicial.

Paso 6. Filtrado de paquetes por la dirección de destino. Uso de ACLs extendidas

Con las ACLs estándar sólo es posible filtrar paquetes por la dirección de origen. Las ACLs extendidas nos permiten realizar el filtrado por la dirección de origen y también por la de destino y por diversos campos de la cabecera IP e incluso de la cabecera ICMP, TCP, UDP, etc.. Las ACLs extendidas tienen la siguiente sintaxis:

```
ACcEss-list n°_lista Permit|DEny protocolo IP_origen [wild-mask]
[operación] [Puerto_origen] IP_destino [wild-mask] [operación]
[Puerto_destino] [established]
```

donde '**n°_lista**' es un valor entre 100 y 199 que identifica la ACL.

Por ejemplo supongamos que queremos descartar en la interfaz LAN de nuestro router todos los paquetes entrantes cuya dirección de origen pertenezca a la red 147.156.0.0/16, y todos los salientes cuya dirección de destino pertenezca a dicha red. Debemos construir dos ACLs diferentes y aplicarlas sobre esa interfaz en sentidos contrarios. La forma de definir las y aplicarlas sería:

```
ACcEss-list 100 DEny IP 147.156.0.0 0.0.255.255 Any
ACcEss-list 100 Permit IP Any Any
ACcEss-list 101 DEny IP Any 147.156.0.0 0.0.255.255
ACcEss-list 101 Permit IP Any Any
INterface FastEthernet 0
IP ACcEss-group 100 In
IP ACcEss-group 101 Out
```

La ACL 100 realizaría un filtrado equivalente al de la ACL 2 (estándar) que vimos en el ejemplo del paso 5, pero en este caso sobre la interfaz LAN. La ACL 101 filtraría los paquetes dirigidos a la red 147.156.0.0/16.

Práctica 5: Filtrado de tráfico en routers

TAREA PARA LOS ALUMNOS

Los alumnos deberán definir dos ACLs extendidas para descartar los paquetes cuya dirección de origen o destino coincida con la del host A de su LAN.

Una vez definidas y aplicadas las ACLs comprobarán mediante el comando ping y la herramienta Ethereal que el router descarta tanto los paquetes ICMP Echo como los Echo-reply cuando se trata del host A.

Una vez terminada esta prueba los alumnos deben **apagar y encender el router** para forzar una carga de la configuración inicial.

Recordemos que el comando `'Show IP ACCEss-lists'` en modo Privilegiado nos permite comprobar las ACLs definidas y obtener información sobre el número de veces que se ha satisfecho cada una de las reglas que componen cada ACL.

Paso 7. Uso de ACLs extendidas para filtrar por el puerto de origen o destino de TCP o UDP

Se pueden utilizar ACLs para filtrar tráfico en función de otros campos de la cabecera IP, tales como el campo protocolo (el que indica si el paquete contiene tráfico TCP, UDP, etc.), o incluso filtrar en base al número de puerto de la cabecera UDP o TCP. De este modo podemos hacer filtrados según el protocolo del nivel de aplicación.

Para probar estas funcionalidades utilizaremos ahora el comando telnet que accede al puerto 23 de TCP. Como el comando telnet permite especificar el número de puerto al que se quiere conectar lo utilizaremos también para acceder al servicio Daytime, que utiliza el puerto 13. El comando telnet siempre funciona sobre TCP.

Supongamos que nuestra LAN es la red 147.156.0.0/16 y que queremos que solo el host 147.156.1.11 pueda conectar a servidores web del exterior (un ejemplo típico de aplicación de esta regla sería el caso en que ese host fuera el servidor proxy y quisiéramos obligar a todos los usuarios a utilizarlo). Las reglas de la correspondiente ACL serían:

```
ACCEss-list 100 Permit Tcp 147.156.1.11 0.0.0.0 Any EQ 80
ACCEss-list 100 DENy Tcp 147.156.0.0 0.0.255.255 Any EQ 80
ACCEss-list 100 Permit IP Any Any
INterface FastEthernet 0
IP ACCEss-group 100 In
```

La primera regla permite los paquetes con dirección de origen 147.156.1.11 cuando el campo protocolo es TCP y el puerto de destino es el 80. La segunda regla descarta los paquetes cuya dirección de origen pertenece a la red 147.156.0.0/16 cuando el campo protocolo es TCP y el puerto de destino es 80. La tercera regla permite que el resto de tráfico de nuestra LAN pueda salir al exterior.

Fijémonos ahora en la sintaxis que utiliza el comando `'ACCEss-lists'` cuando se trata de ACLs extendidas. En primer lugar, detrás de la palabra clave `'Permit'` o `'DENy'` aparece el protocolo, que puede ser IP, ICMP, TCP o UDP (también se contemplan otros protocolos menos utilizados). A continuación aparece la dirección IP de origen seguida de la `'wild-mask'` con lo que se especifica el rango de direcciones de origen al que se aplica dicha regla. Alternativamente se puede usar la palabra clave `'Any'` para indicar cualquier dirección de origen. Después se indica la dirección de destino, bien mediante los dos argumentos (dirección y `'wild-mask'`) o mediante la palabra clave `'Any'`, como antes. Por último la expresión `'EQ 80'` indica que esta regla solo debe aplicarse cuando el puerto de destino sea el 80, ya que en este caso queremos filtrar solo los paquetes dirigidos a servidores HTTP.

Arquitectura de Redes y Servicios

En lugar de especificar los puertos de forma numérica en los casos más habituales se puede indicar el nombre del servicio. Por ejemplo las siguientes reglas son equivalentes a las anteriores:

```
ACcess-list 100 Permit Tcp 147.156.1.11 0.0.0.0 Any EQ WWW
ACcess-list 100 DEny Tcp 147.156.0.0 0.0.255.255 Any EQ WWW
ACcess-list 100 Permit IP Any Any
INterface Fastethernet 0
IP ACCEss-group 100 In
```

También es posible filtrar tráfico en base al puerto de origen, no al puerto de destino. Por ejemplo las siguientes reglas conseguirían el mismo efecto que las anteriores, pero filtrando no los paquetes que envían los clientes sino las respuestas recibidas desde cualquier servidor Web del exterior:

```
Access-list 100 Permit Tcp Any EQ WWW 147.156.1.11 0.0.0.0
ACcess-list 100 DEny Tcp Any EQ WWW Any
ACcess-list 100 Permit IP Any Any
INterface Serial 0
IP ACCEss-group 100 In
```

Obsérvese que en este caso hemos aplicado la ACL sobre el tráfico entrante en la interfaz WAN. Este filtrado sería efectivo pero generaría inútiles intentos de conexión en los servidores cada vez que algún usuario intentara conectar, por lo que sería más correcto hacerlo de la otra forma.

TAREA PARA LOS ALUMNOS

En primer lugar comprobaremos que podemos hacer ping y telnet a través del router entre ordenadores de la LAN norte y la LAN sur. También probaremos a hacer telnet al puerto 13 mediante el comando '**telnet dirección_IP 13**'. De esta forma utilizamos el servicio Daytime por TCP.

Los alumnos deberán ahora configurar en los routers la ACL adecuada para filtrar el tráfico enviado por el host A al puerto 23, es decir deben bloquear el tráfico del host A hacia servidores telnet. Una vez definida y aplicada la ACL comprobarán que el resultado se corresponde con lo esperado.

A continuación deben borrar la ACL anterior y definir una que filtre el tráfico enviado por el host A hacia servidores telnet y daytime (puerto 13). Una vez definida y aplicada comprobarán que el resultado corresponde con lo esperado.

Los alumnos deberán utilizar el comando '**Show IP ACCEss-lists**' para comprobar la correcta definición de las ACLs y para comprobar como el contador de paquetes de cada regla se incrementa según lo esperado.

Una vez terminada esta prueba los alumnos deben **apagar y encender el router** para forzar una carga de la configuración inicial.

También es posible definir reglas para efectuar el filtrado por el número de puerto en datagramas UDP. La sintaxis es la misma que en el caso de TCP, por ejemplo las siguientes reglas bloquearían el tráfico UDP dirigido al puerto 50 de cualquier host en la LAN:

```
ACcess-list 100 DEny Udp Any Any EQ 50
ACcess-list 100 Permit IP Any Any
INterface Fastethernet 0
IP ACCEss-group 100 Out
```

Paso 8. Uso de ACLs extendidas para bloquear el establecimiento de conexiones TCP

En muchos casos se quiere impedir que desde fuera de la LAN se puedan establecer conexiones TCP hacia adentro (salvo probablemente a algunos servidores concretos), pero sin embargo no se quiere restringir la posibilidad de iniciar conexiones TCP desde el interior. La razón es que las conexiones entrantes representan un riesgo de seguridad mucho mayor que las conexiones salientes.

Este tipo de situaciones está previsto en las reglas de las ACLs mediante la palabra clave opcional **ESTABLISHED**, que se coloca al final de la regla. Si aparece esta opción indica que la regla solo se aplica (permitiendo o denegando) cuando el paquete tiene activado el flag ACK o RST. Como el primer paquete enviado al establecer una conexión TCP no tiene activado el flag ACK esta regla no se le aplica. Así por ejemplo si en la interfaz WAN ponemos una ACL para el tráfico entrante que tenga las siguientes reglas:

```
Access-list 100 Permit Tcp Any Any Established
Access-list 100 DENy IP Any Any
Interface Serial 0
IP ACCEss-group 100 In
```

Estamos impidiendo que entre cualquier tráfico que no corresponda a conexiones TCP que se hayan establecido desde dentro. La opción **ESTABLISHED** solo tiene sentido cuando se la combina con dos reglas **Permit** y **DENy**, como se ha hecho en el ejemplo anterior.

Veamos un ejemplo concreto. Supongamos que en nuestra maqueta queremos proteger especialmente el host B. Para ello hemos decidido no permitir en él conexiones TCP entrantes, solo salientes. Además hemos decidido filtrar para dicho host cualquier tráfico no TCP (UDP, ICMP, etc.). Las reglas que nos permitirían configurar estas restricciones en Router_Norte serían las siguientes:

```
Access-list 100 Permit Tcp Any 10.0.1.200 0.0.0.0 Established
Access-list 100 DENy IP Any 10.0.1.200 0.0.0.0
Access-list 100 Permit IP Any Any
Interface Serial 0
IP ACCEss-group 100 In
```

La primera regla está dejando entrar el tráfico TCP de conexiones establecidas. La segunda está denegando todo tráfico hacia 10.0.1.200. La tercera permite cualquier tipo de tráfico. Esta sería una configuración de reglas típica de un cortafuegos. Normalmente cuando se configura un router como cortafuegos se procede filtrando todo el tráfico entrante que no corresponda a conexiones TCP originadas desde dentro, excepto el dirigido a unos pocos hosts, como por ejemplo los servidores, que deben estar accesibles desde el exterior. Eso es lo que harán los alumnos en el ejercicio que se plantea a continuación.

TAREA PARA LOS ALUMNOS

Los alumnos deberán definir en los routers una ACL que impida el establecimiento de conexiones TCP entrantes y cualquier tráfico entrante no TCP dirigido a cualquier ordenador de su LAN, excepto el host B. Una vez configurada y aplicada la ACL comprobarán que efectivamente no es posible comunicar desde fuera con los ordenadores de la LAN, excepto con el host B.

Una vez terminada esta prueba los alumnos deben **apagar y encender el router** para forzar una carga de la configuración inicial.

Aunque sea obvio, vale la pena destacar que los filtros aplicados en routers no pueden controlar los accesos que se realizan entre hosts de una misma LAN. Así por ejemplo todas las ACLs que estamos configurando no imponen ninguna restricción en la comunicación entre HNA y HNB. Para proteger a los

hosts en estas situaciones se necesita aplicar herramientas de control de acceso a nivel de host, tales como los cortafuegos (Iptables en Linux) o los TCP Wrappers.

Paso 9. Uso de ACLs extendidas para bloquear tráfico ICMP

Las ACL extendidas también nos permiten definir reglas que filtren el tráfico ICMP. Por ejemplo la siguiente regla descartaría cualquier mensaje ICMP:

```
Access-list 100 Deny ICMP Any Any
```

En muchos casos los cortafuegos bloquean por completo el intercambio de mensajes ICMP con el exterior. Esto suele causar problemas en el mecanismo de descubrimiento de la MTU del trayecto por el cual muchos TCP ajustan su MSS al valor óptimo, ya que este mecanismo se basa en la recepción de mensajes ICMP ‘Destination Unreachable’ para descubrir ese valor óptimo y evitar los problemas de fragmentación. Cuando un host que utiliza este mecanismo al establecer una conexión TCP deja de recibir el correspondiente mensaje ICMP la conexión no se establece. Por este motivo es bastante habitual, incluso en el caso de filtrar los mensajes ICMP, excluir del filtro los mensajes ‘Destination Unreachable’. Esto se consigue combinando las siguientes reglas:

```
Access-list 100 Permit ICMP Any Any Unreachable  
Access-list 100 Deny ICMP Any Any  
Access-list 100 Permit IP Any Any
```

Estas reglas normalmente se aplicarían en un ACL que actuara sobre el tráfico entrante en la interfaz de conexión a Internet de la LAN de una empresa, por ejemplo.

Suele ser conveniente, aun en el caso de impedir el tráfico ICMP, dejar que los usuarios de la LAN puedan hacer ping al exterior. Esto puede hacerse si se dejan pasar en entrada los mensajes ICMP Echo-reply. Si quisiéramos permitir solo los ping salientes y los ‘Destination Unreachable’ utilizaríamos las siguientes reglas:

```
Access-list 100 Permit ICMP Any Any ECHO-Reply  
Access-list 100 Permit ICMP Any Any Unreachable  
Access-list 100 Deny ICMP Any Any  
Access-list 100 Permit IP Any Any
```

Con estas reglas estamos permitiendo que los usuarios de la LAN hagan ping al exterior, pero no que los del exterior hagan ping a hosts de la LAN, pues no estamos dejando entrar los mensajes ICMP Echo.

TAREA PARA LOS ALUMNOS

Los alumnos deberán definir en los routers una ACL que impida cualquier tráfico ICMP entrante excepto el ‘Destination Unreachable’ y el ‘Echo-Reply’. La ACL deberá actuar sobre tráfico entrante en la interfaz Serie, pero deberá aplicarse en base a la dirección de origen de los paquetes, de forma que solo actúe sobre las direcciones de la LAN remota que se encuentren en la mitad superior del rango (es decir entre la dirección 128 y la 255). Una vez configurada y aplicada la ACL comprobarán que funciona correctamente.

En todos los ejemplos que hemos mostrado hasta aquí el orden de las reglas era siempre importante, ya que si se modificaba cambiaba la semántica de la ACL. Sin embargo en este último ejemplo el orden de las dos primeras reglas es irrelevante, si se permutan el significado de la ACL se mantiene igual. En estos casos es conveniente colocar en primer lugar la regla que se dará con más frecuencia, ya que el router las

Práctica 5: Filtrado de tráfico en routers

evalúa por el orden en que se han definido. Por tanto si ponemos primero la que se da con más frecuencia reducimos el tiempo que empleará el router en analizar la ACL cada vez que tenga que conmutar un paquete.

Paso 10. Finalización

Una vez finalizada la práctica los alumnos deberán realizar las siguientes tareas:

1. Volverán a poner el nombre habitual al fichero de los DNS (en caso de que lo hubiéramos cambiado), comando

```
mv /etc/resolv.conf.old /etc/resolv.conf
```

2. Cerrarán ordenadamente el sistema operativo Linux de los hosts mediante el comando:

```
shutdown -h 0
```

3. Apagarán los routers y las regletas de enchufes que tengan interruptores.
4. Devolverán las conexiones de red de los hosts a las tomas de la pared en las que se encontraban inicialmente, utilizando para ello los latiguillos paralelos.

APÉNDICE

Resumen de los comandos relacionados con ACLs

Definición de una regla en una ACL estándar (modo Configuración):

```
Access-list n°_lista Permit|DEny IP_origen [wild-mask]
```

Donde:

'**n°_lista**' es el identificador de la ACL y está comprendido entre 1 y 99.

'**IP_origen**' es la dirección IP de origen del paquete sobre el que se aplicará la regla. Si la **wild-mask** que le sigue es distinta de 0.0.0.0 especifica un rango

'**wild-mask**' es una máscara que indica con ceros la parte fija y con unos la parte variable de la dirección IP sobre la que se aplicará la regla. Si se omite se supone que es 0.0.0.0, que significa que la regla solo se aplica sobre la dirección indicada.

Definición de una regla en una ACL extendida para tráfico IP (modo Configuración):

```
Access-list n°_lista Permit|DEny IP IP_origen [wild-mask]  
IP_destino [wild-mask]
```

Donde:

'**n°_lista**' es el identificador de la ACL y está comprendido entre 100 y 199.

'**IP_origen**' es la dirección IP de origen del paquete sobre el que se aplicará la regla. Si la '**wild-mask**' que le sigue es distinta de 0.0.0.0 el valor especificado se interpreta como el inicio de un rango

'**wild-mask**' es una máscara que indica con ceros la parte fija y con unos la parte variable de la dirección IP sobre la que se aplicará la regla. Si se omite se supone que es 0.0.0.0, que significa que la regla solo se aplica sobre la dirección indicada.

'**IP_destino**' es la dirección IP de destino del paquete sobre el que se aplicará la regla. Si la '**wild-mask**' que le sigue es distinta de 0.0.0.0 el valor especificado se interpreta como el inicio de un rango

Definición de una regla en una ACL extendida para tráfico TCP o UDP (modo Configuración):

```
Access-list n°_lista Permit|DEny protocolo IP_origen [wild-mask]  
[operación] [Puerto_origen] IP_destino [wild-mask] [operación]  
[Puerto_destino] [Established]
```

Donde:

'**n°_lista**' es el identificador de la ACL y está comprendido entre 100 y 199.

'**protocolo**' indica el valor del campo protocolo del paquete IP. Puede ser Tcp o Udp.

'**IP_origen**' es la dirección IP de origen del paquete sobre el que se aplicará la regla. Si la '**wild-mask**' que le sigue es distinta de 0.0.0.0 el valor especificado se interpreta como el inicio de un rango

'**wild-mask**' es una máscara que indica con ceros la parte fija y con unos la parte variable de la dirección IP sobre la que se aplicará la regla. Si se omite se supone que es 0.0.0.0, que significa que la regla solo se aplica sobre la dirección indicada.

Práctica 5: Filtrado de tráfico en routers

'**operación**' es un operador relacional que se aplica entre el valor en el campo puerto de origen de la cabecera TCP o UDP y el valor especificado en el campo siguiente, '**Puerto_origen**' o '**Puerto_destino**'

'**Puerto_origen**' especifica con que valor se va a comparar el campo puerto origen de la cabecera TCP o UDP.

'**IP_destino**' es la dirección IP de destino del paquete sobre el que se aplicará la regla. Si la '**wild-mask**' que le sigue es distinta de 0.0.0.0 el valor especificado se interpreta como el inicio de un rango

'**Puerto_destino**' especifica con que valor se va a comparar el campo puerto destino de la cabecera TCP o UDP.

'**Established**' indica que la regla se aplicará únicamente a los segmentos TCP que lleven puesto el flag ACK

Definición de una regla en una ACL extendida para tráfico ICMP (modo Configuración):

```
Access-list n°_lista Permit|Deny ICMP IP_origen [wild-mask]
IP_destino [wild-mask] [tipo]
```

Donde:

'**n°_lista**' es el identificador de la ACL y está comprendido entre 100 y 199.

'**IP_origen**' es la dirección IP de origen del paquete sobre el que se aplicará la regla. Si la '**wild-mask**' que le sigue es distinta de 0.0.0.0 el valor especificado se interpreta como el inicio de un rango

'**wild-mask**' es una máscara que indica con ceros la parte fija y con unos la parte variable de la dirección IP sobre la que se aplicará la regla. Si se omite se supone que es 0.0.0.0, que significa que la regla solo se aplica sobre la dirección indicada.

'**IP_destino**' es la dirección IP de destino del paquete sobre el que se aplicará la regla. Si la '**wild-mask**' que le sigue es distinta de 0.0.0.0 el valor especificado se interpreta como el inicio de un rango

'**tipo**' especifica el tipo de mensaje ICMP sobre el que se aplica la regla

Aplicación de una ACL en una interfaz (modo Configuración de Interfaz):

```
IP Access-group n°_lista In|Out
```

El operando In|Out indica sentido entrante o saliente.

Desaplicar una ACL sobre una interfaz (modo Configuración de Interfaz):

```
NO IP Access-group n°_lista
```

Borrado de una ACL definida (modo Configuración):

```
NO Access-list n°_lista
```

Ver las ACLs definidas:

```
SHOW IP Access-lists
```