UD4. ACL

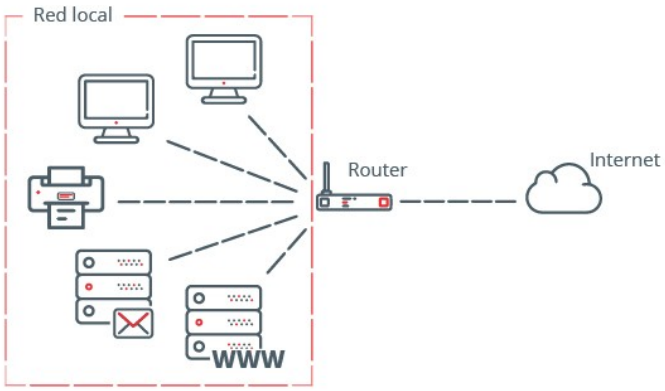
Índice

Firewall y DMZ

La seguridad de una red informática es uno de los pilares básicos que hay que tener en cuenta de cara a su análisis, su diseño y su posterior implementación y mantenimiento.

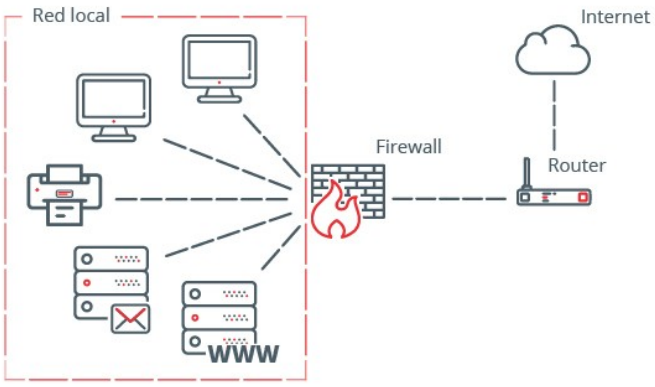
Cuando se permite el acceso desde Internet a una página web, servidor de correo, servidor de ficheros, red privada virtual, etc., aumenta el riesgo de sufrir un incidente de seguridad. Si un ciberdelincuente consigue vulnerar la seguridad de uno de estos servidores, podría comprometer el resto de dispositivos conectados a la red, incluso aquellos que no son accesibles desde Internet. Un acceso no deseado podría derivar en una infección por ransomware, comunicaciones espiadas, ficheros robados, caídas de servicio, etc.

Una configuración errónea de una red en una organización que cuenta con un servidor de correo y un servidor web, sería la siguiente.



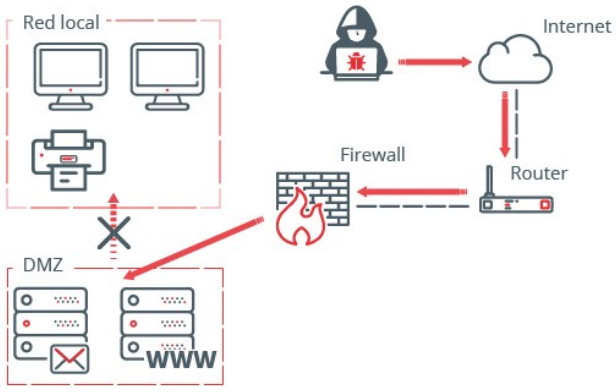
Son muchas las técnicas que suelen utilizarse para este fin, pero, sin duda, una de las más importantes son las listas de control de acceso (ACL).

El administrador de red debe conocer a fondo su manejo y configuración para proteger las redes de un uso no autorizado. En un router Cisco, se puede configurar un firewall simple, que proporcione capacidades básicas de filtrado de tráfico mediante ACL. Los administradores utilizan las ACL para detener el tráfico o para permitir solamente tráfico específico en sus redes.

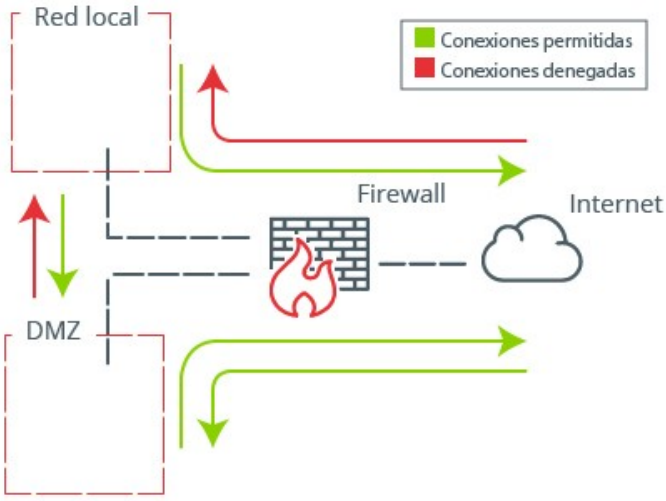


Una zona desmilitarizada es una **red aislada que se encuentra dentro de la red interna de la organización**. En ella se encuentran ubicados exclusivamente todos los recursos de la empresa que deben ser accesibles desde Internet, como el servidor web o de correo. Un ejemplo de una red local con una DMZ se ve en la página siguiente.

Por lo general, una DMZ permite las conexiones procedentes tanto de Internet, como de la red local de la empresa donde están los equipos de los trabajadores, pero las conexiones que van desde la DMZ a la red local, no están permitidas. Esto se debe a que los servidores que son accesibles desde Internet son más susceptibles a sufrir un ataque que pueda comprometer su seguridad. Si un ciberdelincuente comprometiera un servidor de la zona desmilitarizada, tendría muchos más complicado acceder a la red local de la organización, ya que las conexiones procedentes de la DMZ se encuentran bloqueadas.



Para configurar una zona desmilitarizada en la red de la organización, es necesario contar con un cortafuegos o *firewall*. Este dispositivo, será el encargado de segmentar la red y permitir o denegar las conexiones. En la siguiente tabla, de manera somera, se muestra el tipo de conexiones recomendables que permitiría o denegaría el *firewall* dependiendo su origen y destino:



Origen	Destino	Política
Internet	DMZ	Permitido
Internet	LAN	Denegado
DMZ	Internet	Permitido
DMZ	LAN	Denegado
LAN	DMZ	Permitido
LAN	Internet	Permitido

ACL

Se puede decir que una ACL es un conjunto de instrucciones o sentencias que permiten (permit) o deniegan (deny) un determinado tráfico de paquetes en la red corporativa.

La máscara de red Wildcard

Cada entrada de una ACL incluye el uso de una máscara Wildcard o "comodín". Una máscara Wildcard es una cadena de 32 dígitos binarios que el router utiliza para determinar qué bits de la dirección del paquete debe examinar para obtener una coincidencia.

Como ocurre con las máscaras de subred, los números 1 y 0 en la máscara Wildcard identifican lo que hay que hacer con los bits de dirección IP correspondientes. Sin embargo, en una máscara Wildcard, estos bits se utilizan para fines diferentes y siguen diferentes reglas:

- Bit 0 : establece la coincidencia con el valor del bit correspondiente en la dirección IP.
- Bit 1 : omite el valor del bit correspondiente en la dirección IP.

Mientras que las máscaras de subred utilizan 1 y 0 binarios para identificar la red, la subred y la porción del host de una dirección IP, las máscaras Wildcard se utilizan para filtrar direcciones IP individuales o grupos de ellas, permitiendo o denegando el acceso a los recursos.

A las máscaras Wildcard a menudo se les denomina "máscaras inversas". La razón es que, a diferencia de una máscara de subred en la que el 1 binario equivale a una coincidencia y el 0 binario no, en las máscaras Wildcard es al revés. Por ejemplo:

Podemos obtener la máscara wildcard restando a 255.255.255.255 menos la máscara IP normal. La máscara 0.0.0.0 refiere por tanto un host.

Dirección IP	Máscara de red	Máscara *Wildcard*
192.168.1.0	*255.255.255.0*	*0.0.0.255*
172.16.192.0	*255.255.255.240*	*0.0.0.15*

Mostrar ACL

show access-lists
show running-config | include access-list numero

Comentar ACL

Puedes utilizar la palabra clave remark para incluir comentarios (remarks) sobre entradas en cualquier ACL de IP estándar o extendida. Estos comentarios facilitan la comprensión y la revisión de las ACL. Cada línea de comentarios tiene un límite de 100 caracteres.

El comentario puede ir antes o después de una instrucción permit o deny. Debe ser coherente en cuanto a la ubicación del comentario, de manera que quede claro qué comentario describe cuál de las instrucciones permit o deny. Por ejemplo, sería confuso colocar algunos comentarios antes de las instrucciones permit o deny correspondientes y otros después de estas.

access-list numero|nombre remark comentario

Tipos de ACL

Una ACL controla el acceso a una red mediante el análisis de los paquetes entrantes y salientes y la transferencia o el descarte de estos según determinados criterios, como la dirección IP de origen, la dirección IP de destino o alguno de los protocolos incluidos en el paquete.

Un router que define y activa una ACL filtra los paquetes utilizando "reglas" para determinar si permite o deniega el tráfico. El filtrado de paquetes se puede realizar en la capa 3, la capa de red, o en la capa 4, la capa de transporte filtrando paquetes según puerto de origen/destino del segmento TCP/UDP.

Por lo general, existen dos tipos clásicos de ACL:

1. ACL estándar. Permite el filtrado de paquetes de datos **únicamente verificando la dirección IP de origen**. Son el tipo más antiguo y controlan el tráfico por la comparación de la dirección de origen de los paquetes IP con las direcciones configuradas en la ACL. Las ACL estándar están identificadas por el número que se les ha asignado, que puede variar entre 1 y 99, o entre 1300 y 1999. Su formato es el siguiente:

access-list acl-number {permit | deny} host | origen [Wildcard-origen | any]

Ejemplo:

access-list 10 permit 192.168.148.0 0.0.1.255
access-list 10 permit host 192.168.10.0
no access-list 10

2. ACL extendidas. Filtran el tráfico no sólo según la dirección IP de origen, sino también según la dirección IP de destino, el protocolo y los números de puertos. Con frecuencia son más empleadas que las ACL estándar, porque son más específicas y ofrecen un mayor control. El rango de números de las ACL extendidas va de 100 a 199 y de 2000 a 2699. Los formatos son los siguientes:

access-list acl-number {permit | deny | remark} protocolo origen [Wildcard-origen] [operación] [puerto origen o nombre] destino [Wildcard-destino] [operación] [puerto destino o nombre] [established]

- acl-number: identifica el número de lista de acceso.
- protocolo: IP, TCP, UDP, ICMP, GRE, IGRP
- origen / destino: identificadores de direcciones IP origen y destino.
- Wildcard-origen/Wildcard destino: máscaras de Wildcard. También podemos poner **host** si es un host (ej. 10.1.1.2 0.0.0.0 es lo mismo que "host 10.1.1.2") o **any** si es cualquiera
- Operación: lt, gt, eq, neq.
- Puerto destino: número de puerto o nombre del protocolo si es conocido, Podemos mirar la lista aquí. Por ejemplo para TCP:
    <0-65535> Port number
    ftp File Transfer Protocol (21)
    pop3 Post Office Protocol v3 (110)
    smtp Simple Mail Transport Protocol (25)
    telnet Telnet (23)
    www World Wide Web (HTTP, 80)
- **Established**: permite que pase el tráfico TCP si el paquete utiliza una conexión preestablecida (solo para el protocolo TCP). Sin este parámetro los clientes pueden enviar tráfico a un servidor web, pero no recibir el tráfico que vuelve de dicho servidor.

Ejemplos:

access-list 101 permit ip 10.1.1.0 0.0.0.255 172.16.1.0 0.0.0.255
access-list 101 permit tcp host 10.1.1.2 host 172.16.1.1
access-list 101 permit tcp host 10.1.1.2 host 172.16.1.1 eq telnet
access-list 101 permit tcp host 10.1.1.2 host 172.16.1.1 eq 23
show access-list

Tanto a las ACL estándar como extendidas es posible referenciarlas mediante un nombre descriptivo en lugar de un número, lo que se conoce como ACL nombradas. Existen, además, otros tipos de ACL enfocados en propósitos específicos de configuración y manejo del filtrado de los paquetes de datos, como son las ACL dinámicas, reflexivas, basadas en tiempo y basadas en el contexto, entre otras.

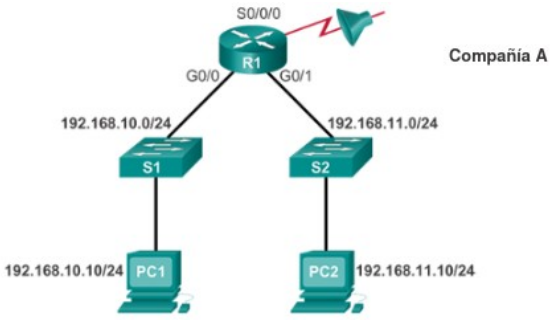
ACL con nombre

ip access-list [standard | extended] nombre

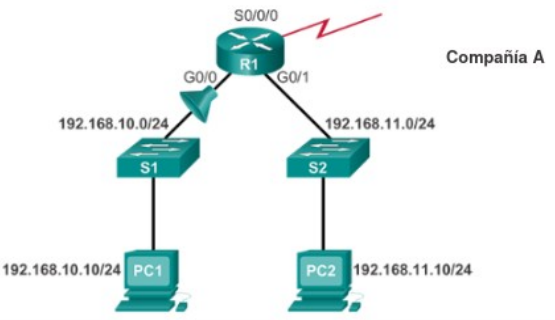
una vez definida la lista pongo las reglas según esta sintaxis que es la misma que hemos visto antes pero sin poner access-list y su número antes. Por ejemplo en el caso de una estándar sería:

{permit | deny | remark} host | origen [Wildcard-origen | any]

Puedes definir las ACL y seguir sin aplicarlas. Pero, las ACL no tienen ningún efecto hasta que se aplican a la interfaz del router. Una buena práctica sería aplicar el ACL en la interfaz más cerca al origen del tráfico. Las estándar se suelen aplicar más cerca del destino y las extendidas más cerca del origen. Una lista de acceso tiene una deny ip any any implícitamente al final de cualquier lista de acceso. Si el tráfico está relacionado con una solicitud DHCP y si no se permite explícitamente, el tráfico se descarta porque cuando observa la solicitud DHCP en IP, la dirección de origen es s=0.0.0.0 (Ethernet1/0), d=255.255.255.255, len 604, rcvd 2 UDP src=68, dst=67. Observe que la dirección IP de origen es 0.0.0.0 y la dirección de destino es 255.255.255.255. El puerto de origen es 68 y el puerto de destino es 67. Por lo tanto, debe permitir este tipo de tráfico en su lista de acceso o de lo contrario el tráfico se descarta debido a una negación implícita al final de la sentencia.




```
R1(config)#no access-list 1
R1(config)#access-list 1 deny host 192.168.10.10
R1(config)#access-list 1 permit 192.168.10.0 0.0.0.255
R1(config)#interface s0/0/0
R1(config-if)#ip access-group 1 out
```




```
R1(config)#no access-list 1
R1(config)#access-list 1 deny host 192.168.10.10
R1(config)#access-list 1 permit any
R1(config)#interface g0/0
R1(config-if)#ip access-group 1 in
```

Configuración de ACL

La configuración de ACL pasa por dos etapas claramente diferenciadas. En primer lugar, hay que definir la regla de filtrado, ya sea utilizando el tipo estándar o el tipo extendido y, en segundo lugar, aplicar el filtro sobre una interfaz concreta. A la hora de aplicar una ACL hay que tener en cuenta varios aspectos clave:

1. **Sobre qué interfaz**. Las listas de acceso se deben colocar lo más cerca posible del origen.
2. **En qué sentido:** Entrante (in), cuando el tráfico llega a la interfaz y luego pasa hacia el router, o saliente (out), cuando el tráfico ha pasado por el router y sale por la interfaz. La sintaxis para asociar una ACL es la siguiente:

   ip access-group acl-number {in | out}

   Solo se puede especificar una ACL por protocolo y por interfaz. Si la ACL es entrante, se comprueba al recibir el paquete, y si es saliente, se comprueba después de recibir y enrutar el paquete en la interfaz saliente.

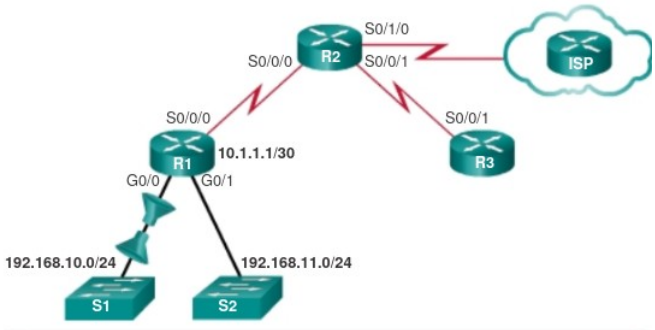
3. **El orden de aplicación de las reglas ACL**. El IOS del router comprueba la coincidencia de cada paquete en el mismo orden en que se crearon las ACL y, una vez que se verifica una coincidencia, no se siguen verificando el resto de reglas ACL. El tráfico que entra en el router se compara con las entradas de ACL según el orden en que ocurren las entradas en el router. Se agregan nuevas declaraciones al final de la lista. El router continúa mirando hasta obtener una coincidencia. Si no se encuentran coincidencias cuando el router llega al final de la lista, se rechaza el tráfico. Por este motivo, debe tener las entradas que se consultan con frecuencia al principio de la lista. Hay una negación implícita para el tráfico que no se permite. Una ACL de una sola entrada con una sola entrada denegada puede denegar todo el tráfico. Debemos tener por lo menos una declaración de permiso en una ACL o se bloquea todo el tráfico.

Recuerda que si no hay coincidencia con ninguna regla ACL, se ejecuta por defecto una sentencia implícita 'deny any', que no permitirá que ningún paquete que no coincida con alguna de las ACL definidas sea aceptado.

Por ejemplo, un router desea permitir todos los paquetes que provengan de la red 10.0.0.0/8 y salgan por su interfaz Fa0/2. La definición y aplicación de ACL sería así:

```
Router> enable
Router# configure terminal
Router(config)# access-list 1 permit 10.0.0.0 0.255.255.255
Router(config)# interface Fastethernet 0/2
Router(config-if)# ip access-group 1 out
```

Una ACL extendida comúnmente se debería aplicar cerca del origen. En esta topología, la interfaz más cercana al origen del tráfico de destino es la interfaz G0/0. La solicitud de tráfico web de los usuarios en la LAN 192.168.10.0/24 entra a la interfaz G0/0. El tráfico de retorno de las conexiones establecidas a los usuarios en la LAN sale de la interfaz G0/0. En el ejemplo, se aplica la ACL a la interfaz G0/0 en ambos sentidos. La ACL de entrada, 103, revisa el tipo de tráfico. La ACL de salida, 104, revisa si hay tráfico de retorno de las conexiones establecidas. Esto restringe el acceso a Internet desde 192.168.10.0 para permitir solamente la navegación de sitios web.




```
R1(config)#access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 80
R1(config)#access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1(config)#access-list 104 permit tcp any 192.168.10.0 0.0.0.255 established
R1(config)#interface g0/0
R1(config-if)#ip access-group 103 in
R1(config-if)#ip access-group 104 out
```

Edición de ACL

Si eliminamos una línea específica de una ACL numerada existente se elimina la ACL entera. Para evitarlo tenemos que configurar la ACL con **ip access-list {standard | extended} {numero | nombre}**

Por nombre ejemplo:

```
router#configure terminal
router(config)#ip access-list extended test
router(config-ext-nacl)#permit ip host 10.2.2.2 host 10.3.3.3
router(config-ext-nacl)#permit tcp host 10.1.1.1 host 10.5.5.5 eq www
router(config-ext-nacl)#permit icmp any any
```

Entro a configurar la lista de ACL

Añado entradas

```
router#show access-list
Extended IP access list test
    permit ip host 10.2.2.2 host 10.3.3.3
    permit tcp host 10.1.1.1 host 10.5.5.5 eq www
    permit icmp any any
```

Todas las eliminaciones se quitan de la ACL y todas las adiciones se realizan al final de la ACL.

```
router#configure terminal
router(config)#ip access-list extended test
router(config-ext-nacl)#no permit icmp any any
router(config-ext-nacl)#permit tcp host 10.4.4.4 host 10.8.8.8
```

Entro a configurar la lista de ACL

Borro una entrada

añado una entrada

```
router#show access-list
Extended IP access list test
    permit ip host 10.2.2.2 host 10.3.3.3
    permit tcp host 10.1.1.1 host 10.5.5.5 eq www
    permit udp host 10.6.6.6 10.10.10.0 0.0.0.255 eq domain
    permit tcp host 10.4.4.4 host 10.8.8.8
```

Por numero ejemplo:

```
Router(config)#access-list 101 permit tcp any any
Router(config)#access-list 101 permit udp any any
Router(config)#access-list 101 permit icmp any any
```

Creo la ACL numerada 101

```
Router#show access-list
Extended IP access list 101
    10 permit tcp any any
    20 permit udp any any
    30 permit icmp any any
```

Las entradas tienen un número de secuencia (en este caso 10,20 y 30)

Agregamos la entrada para la lista de acceso 101 con el número de secuencia 5.

```
Router#configure terminal
Router(config)#ip access-list extended 101
Router(config-ext-nacl)#5 deny tcp any any eq telnet
Router(config-ext-nacl)#no permit udp any any
Router(config-ext-nacl)#exit
```

Entro a configurar la lista de ACL

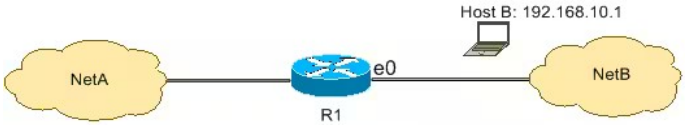
Añado entrada a la lista

También valdría **no 20**

```
Router#show access-list
Extended IP access list 101
    5 deny tcp any any eq telnet
    10 permit tcp any any
    20 permit udp any any
    30 permit icmp any any
```

Ejemplos

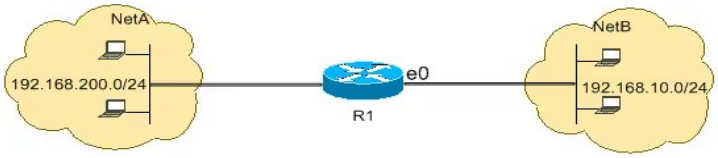
Se permite todo el tráfico con origen en el Host B y destino en la Red A, y se niega el resto del tráfico con origen en la Red B y destino en la Red A.




```
Access-list 1 permit 192.168.10.1
interface e0
ip access-group 1 in
```

Se niega el tráfico originado en el Host B destinado a la Red A, mientras que se permite el resto del tráfico de la Red B para acceder a la Red A.

```
Access-list 1 deny host 192.168.10.1
access-list 1 permit any
ip access-group 1 in
```



Permite que los paquetes IP con un encabezado IP que tengan una dirección de origen en la red 192.168.10.0/24 y una dirección de destino en la red 192.168.200.0/24 accedan a la Red A.

```
Access-list 101 permit 192.168.10.0 0.0.0.255 192.168.200.0 0.0.0.255
ip access-group 101 in
```

Para cumplir con los requisitos de mayor seguridad, inhabilita el acceso Telnet a la red privada desde la red pública. deniega el tráfico Telnet de la Red B (pública) destinado a la Red A (privada), lo que permite a la Red A iniciar y establecer una sesión Telnet con la Red B mientras se permite el resto del tráfico IP.

```
Interface e0
ip access-group 102 in
access-list 102 deny tcp any any eq 23
access-list 102 permit ip any any
```

Sólo las redes internas pueden iniciar una sesión TCP, se permite el tráfico TCP con origen en la Red A y destino en la Red B, mientras que se niega el tráfico TCP de la Red B con destino en la Red A.

```
Interface e0
ip access-group 102 in
access-list 102 permit tcp any any gt 1023 established
```

permite el ICMP originado en la Red A y destinado a la Red B, y se deniegan los pings originados en la Red B y destinados a la Red A.

```
interface ethernet0
ip access-group 102 in
access-list 102 permit icmp any any echo-reply
```

Permitir dns

```
interface ethernet0
ip access-group 102 in
access-list 102 permit udp any any eq domain
access-list 102 permit udp any eq domain any
access-list 102 permit tcp any any eq domain
access-list 102 permit tcp any eq domain any
```

Permitir actualizaciones de enrutamiento
protocolo de información de enrutamiento (RIP):

access-list 102 permit udp any any eq rip

el protocolo de enrutamiento de gateway interior (IGRP):

access-list 102 permit igrp any any

permitir el IGRP mejorado (EIGRP):

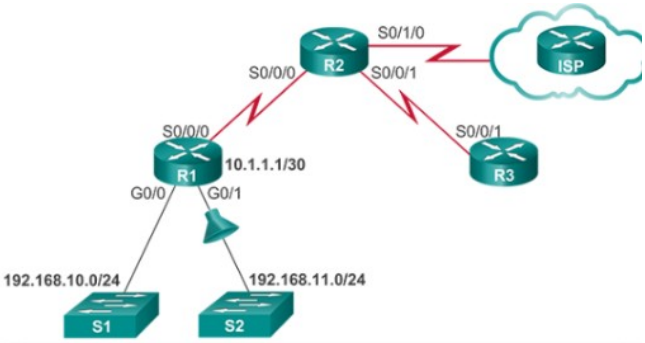
access-list 102 permit eigrp any any

primero la ruta más corta (OSPF):

access-list 102 permit ospf any any

Introduzca este comando para permitir el protocolo de gateway fronterizo (BGP):

access-list 102 permit tcp any any eq 179
access-list 102 permit tcp any eq 179 any

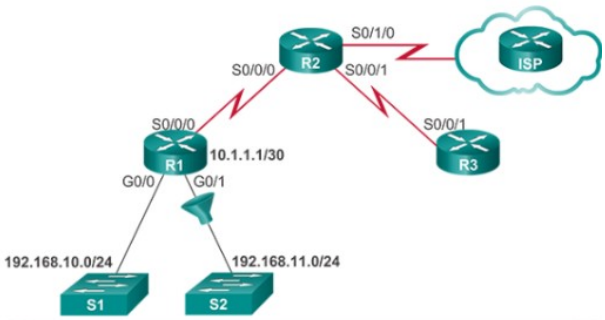



```
R1(config)# access-list 101 deny tcp 192.168.11.0 0.0.0.255
192.168.10.0 0.0.0.255 eq ftp
R1(config)# access-list 101 deny tcp 192.168.11.0 0.0.0.255
192.168.10.0 0.0.0.255 eq ftp-data
R1(config)# access-list 101 permit ip any any
R1(config)# interface g0/1
R1(config-if)#ip access-group 101 in
```

se deniega el tráfico FTP de la subred 192.168.11.0 que va a la subred 192.168.10.0, pero se permite el resto del tráfico. Observe el uso de las máscaras wildcard y de la instrucción deny any explícita. Recuerde que FTP utiliza los puertos TCP 20 y 21, por lo tanto, la ACL requiere ambas palabras claves de nombre de puerto ftp y ftp-data o eq 20 y eq 21 para denegar el tráfico FTP.

Si se utilizan números de puerto en vez de nombres de puerto, los comandos se deben escribir de la siguiente forma:
access-list 114 permit tcp 192.168.20.0 0.0.0.255 any eq 20
access-list 114 permit tcp 192.168.20.0 0.0.0.255 any eq 21

Para evitar que la instrucción deny any implícita al final de la ACL bloquee todo el tráfico, se agrega la instrucción permit ip any any. Si no hay por lo menos una instrucción permit en una ACL, todo el tráfico en la interfaz donde se aplicó esa ACL se descarta. La ACL se debe aplicar en sentido de entrada en la interfaz G0/1 para filtrar el tráfico de la LAN 192.168.11.0/24 cuando ingresa a la interfaz del router.

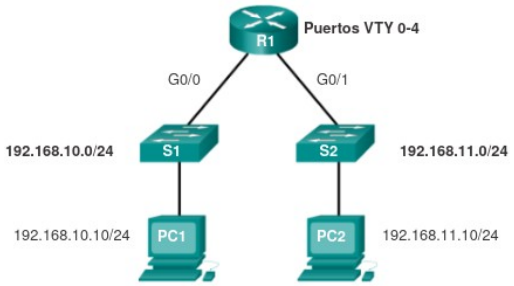



```
R1(config)# access-list 102 deny tcp any 192.168.11.0 0.0.0.255
eq 23
R1(config)# access-list 102 permit ip any any
R1(config)# interface g0/1
R1(config-if)# ip access-group 102 out
```

se deniega el tráfico de Telnet de cualquier origen a la LAN 192.168.11.0/24, pero se permite el resto del tráfico IP. Debido a que el tráfico destinado a la LAN 192.168.11.0/24 sale de la interfaz G0/1, la ACL se aplica a G0/1 con la palabra clave out. Observe el uso de las palabras clave any en la instrucción permit. Esta instrucción permit se agrega para asegurar que no se bloquee ningún otro tipo de tráfico.

Filtrado VTY

Cisco recomienda utilizar SSH para las conexiones administrativas a los routers y switches. Se puede mejorar la seguridad de las líneas administrativas mediante la restricción del acceso a VTY. La restricción del acceso a VTY es una técnica que permite definir las direcciones IP a las que se les permite acceder por Telnet o ssh al proceso de EXEC del router. Puede controlar qué estación de trabajo administrativa o qué red administra el router mediante la configuración de una ACL e instrucción access-class en las líneas VTY.




```
R1(config)#line vty 0 4
R1(config-line)#login local
R1(config-line)#transport input ssh
R1(config-line)#access-class 21 in
R1(config-line)#exit
R1(config)#access-list 21 permit 192.168.10.0 0.0.0.255
R1(config)#access-list 21 deny any
```

El comando access-class configurado en el modo de configuración de línea restringe las conexiones de entrada y salida entre una VTY determinada (en un dispositivo de Cisco)
Las listas de control de acceso estándar y extendidas se aplican a los paquetes que se transportan a través de un router, no están diseñadas para bloquear los paquetes que se originan en el router. Una ACL extendida para Telnet de salida no evita las sesiones de Telnet iniciadas por el router de manera predeterminada.

Por lo general, se considera que el filtrado del tráfico de Telnet o SSH es una función de una ACL de IP extendida, porque filtra un protocolo de nivel superior. Sin embargo, debido a que se utiliza el comando access-class para filtrar sesiones de Telnet/SSH entrantes o salientes por dirección de origen, se puede utilizar una ACL estándar.

La sintaxis del comando access-class es la siguiente (solo se pueden usar numeradas):

Router (config)# access-class número-lista-acceso { in | out }

El parámetro in limita las conexiones de entrada entre las direcciones en la lista de acceso y el dispositivo de Cisco, mientras que el parámetro out limita las conexiones de salida entre un dispositivo de Cisco en particular y las direcciones en la lista de acceso.

Estadísticas

El comando **show access-lists** muestra las estadísticas para cada instrucción que tiene coincidencias. Tanto las instrucciones permit como las deny realizan un seguimiento de las estadísticas de coincidencias; sin embargo, recuerde que cada ACL tiene una instrucción deny any implícita como última instrucción. Esta instrucción no aparece en el comando show access-lists, por lo que no se muestran estadísticas para esa instrucción.

Se pueden borrar los contadores mediante el comando **clear access-list counters**. Este comando se puede utilizar solo o con el número o el nombre de una ACL específica. Como se muestra en la figura 2, este comando borra los contadores de estadísticas para una ACL.