Configurar y filtrar listas de acceso IP

Contenido

Introducción

Este documento describe varios tipos de listas de control de acceso IP (ACL) y cómo pueden filtrar el tráfico de red.

Prerequisites

Requirements

No hay requisitos previos específicos para este documento. Los conceptos descritos están presentes en Cisco IOS®Software Releases 8.3 o posterior. Esto se observa debajo de cada función de lista de acceso.

Componentes Utilizados

En este documento, se analizan varios tipos de ACL. Algunos de estos están presentes porque el Cisco IOS Software, versión 8.3 y otras se introdujeron en versiones de software posteriores. Esto se observa en el análisis de cada tipo.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

Convenciones

Consulte el documento Cisco Technical Tips Conventions (Convenciones sobre consejos técnicos de Cisco) para obtener más información sobre las convenciones de los documentos.

Antecedentes

En este documento, se describe cómo las listas de control de acceso (ACL) IP pueden filtrar el tráfico de la red. También contiene descripciones breves de los tipos de ACL de IP, de la disponibilidad de la función y un ejemplo de uso en una red.

> **Nota:** RFC 1700 contiene números asignados de puertos conocidos.RFC 1918 contiene la asignación de direcciones para redes Internet privadas, direcciones IP que normalmente no deben verse en Internet.

> **Nota:** Solo los usuarios registrados de Cisco pueden acceder a la información interna.

> **Nota:** Las ACL también se pueden utilizar para definir el tráfico a la traducción de direcciones de red (NAT), cifrar o filtrar protocolos no IP como AppleTalk o IPX. El análisis de estas funciones no se cubre en este documento.

Conceptos de ACL

Máscaras

Las máscaras se utilizan con direcciones IP en ACL IP para especificar lo que se debe permitir y denegar. Las máscaras para configurar direcciones IP en las interfaces comienzan con 255 y tienen los valores grandes en el lado izquierdo; por ejemplo, dirección IP 10.165.202.129 con una máscara 255.255.255.224. Las máscaras para las ACL IP son al revés, por ejemplo, la máscara 0.0.0.255. A veces se denomina máscara inversa o máscara comodín. Cuando el valor de la máscara se divide en binarios (0s y 1s), los resultados determinan qué bits de dirección deben tenerse en cuenta al procesar el tráfico. Un 0 indica que los bits de dirección se deben considerar (coincidencia exacta); un 1 en la máscara es un *no importa*. En esta tabla, se explica más el concepto.

	Ejemplo de Máscara
dirección de red (tráfico que se procesará)	*10.1.1.0*
máscara	*0.0.0.255*
dirección de red (binaria)	*00001010.00000001.00000001.00000000*
máscara (binaria)	*00000000.00000000.00000000.11111111*

Según la máscara binaria, puede ver que los primeros tres conjuntos (octetos) deben coincidir exactamente con la dirección de red binaria dada (00001010.00000001.00000001). El último conjunto de números es *no importa* (.11111111). Por lo tanto, todo el tráfico que comienza con 10.1.1 coincide ya que el último octeto es *no importa*. Por lo tanto, con esta máscara, se procesan direcciones de red de 10.1.1.1 a 10.1.1.255 (10.1.1.x).

Reste la máscara normal de 255.255.255.255 para determinar la máscara inversa de ACL. En este ejemplo, la máscara inversa se determina para la dirección de red 172.16.1.0 con una máscara normal 255.255.255.0.

- 255.255.255.255 - 255.255.255.0 (máscara normal) = 0.0.0.255 (máscara inversa)

Observe los equivalentes de ACL.

- El origen/comodín de 0.0.0.0/255.255.255.255 significa **cualquiera.**
- El origen/comodín de 10.1.1.2/0.0.0.0 es el mismo que el **host 10.1.1.2**.

Resumen de ACL

> **Nota:** Las máscaras de subred también pueden representarse como anotaciones de longitud fija. Por ejemplo, 192.168.10.0/24 representa 192.168.10.0 255.255.255.0.

En esta lista, se describe cómo resumir un rango de redes en una sola red para la optimización de ACL. Considere estas redes.

```
192.168.32.0/24
192.168.33.0/24
192.168.34.0/24
192.168.35.0/24
192.168.36.0/24
192.168.37.0/24
192.168.38.0/24
192.168.39.0/24
```

Los primeros dos octetos y el último octeto son iguales para cada red. En esta tabla, hay una

explicación de cómo resumir estas redes en una sola red.

El tercer octeto para las redes anteriores se puede escribir como se ve en esta tabla, correspondiente a la posición del bit del octeto y al valor de dirección para cada bit.

Decimal	128	64	32	16	8	4	2	1
32	0	0	1	0	0	0	0	0
33	0	0	1	0	0	0	0	1
34	0	0	1	0	0	0	1	0
35	0	0	1	0	0	0	1	1
36	0	0	1	0	0	1	0	0
37	0	0	1	0	0	1	0	1
38	0	0	1	0	0	1	1	0
39	0	0	1	0	0	1	1	1
	M	M	M	M	M	D	D	D

Dado que los primeros cinco bits coinciden, las ocho redes anteriores se pueden resumir en una red (192.168.32.0/21 o 192.168.32.0 255.255.248.0). Las ocho combinaciones posibles de los tres bits de orden inferior son relevantes para los rangos de redes en cuestión. Este comando define una ACL que permite esta red. Si usted resta 255.255.248.0 (máscara normal) de 255.255.255.255, es resultado es 0.0.7.255.

```
access-list acl_permit permit ip 192.168.32.0 0.0.7.255
```

Considere este conjunto de redes para una explicación adicional.

```
192.168.146.0/24
192.168.147.0/24
192.168.148.0/24
192.168.149.0/24
```

Los primeros dos octetos y el último octeto son iguales para cada red. En esta tabla, hay una explicación de cómo resumir esto.

El tercer octeto para las redes anteriores se puede escribir como se ve en esta tabla, correspondiente a la posición del bit del octeto y al valor de dirección para cada bit.

Decimal	128	64	32	16	8	4	2	1
146	1	0	0	1	0	0	1	0
147	1	0	0	1	0	0	1	1
148	1	0	0	1	0	1	0	0
149	1	0	0	1	0	1	0	1
	M	M	M	M	M	?	?	?

A diferencia del ejemplo anterior, usted no puede resumir estas redes en una sola red. Si se resumen en una sola red, se convierten en 192.168.144.0/21 porque hay cinco bits similares en el tercer octeto. Esta red resumida 192.168.144.0/21 cubre un rango de redes de 192.168.144.0 a 192.168.151.0. Entre estas, 192.168.144.0, 192.168.145.0, 192.168.150.0 y 192.168.151.0 no se encuentran en la lista dada de cuatro redes. Para cubrir las redes específicas en cuestión, necesita un mínimo de dos redes resumidas. Las cuatro redes dadas se pueden resumir en estas dos redes:

- Para las redes 192.168.146.x y 192.168.147.x, todos los bits coinciden excepto el último, que es un *no importa*. Esto puede escribirse como 192.168.146.0/23 (o 192.168.146.0

255.255.254.0).
- Para las redes 192.168.148.x y 192.168.149.x, todos los bits coinciden excepto el último, que es un *no importa*. Esto puede escribirse como 192.168.148.0/23 (o 192.168.148.0 255.255.254.0).

Este resultado define una ACL resumida para las redes anteriores.

```
!--- This command is used to allow access access for devices with IP
!--- addresses in the range from 192.168.146.0 to 192.168.147.254. access-list 10 permit
192.168.146.0 0.0.1.255


!--- This command is used to allow access access for devices with IP
!--- addresses in the range from 192.168.148.0 to 192.168.149.254 access-list 10 permit
192.168.148.0 0.0.1.255
```

Proceso de 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. Usted debe tener por lo menos una declaración de permiso en una ACL o se bloquea todo el tráfico. Estas dos ACL (101 y 102) tienen el mismo efecto.

```
!--- This command is used to permit IP traffic from 10.1.1.0
!--- network to 172.16.1.0 network. All packets with a source
!--- address not in this range will be rejected. access-list 101 permit ip 10.1.1.0 0.0.0.255
172.16.1.0 0.0.0.255


!--- This command is used to permit IP traffic from 10.1.1.0
!--- network to 172.16.1.0 network. All packets with a source
!--- address not in this range will be rejected. access-list 102 permit ip 10.1.1.0 0.0.0.255
172.16.1.0 0.0.0.255
access-list 102 deny ip any any
```

En el siguiente ejemplo, la última entrada es suficiente. No necesita las tres primeras entradas porque IP incluye TCP, protocolo de datagramas de usuario (UDP) y protocolo de mensajes de control de Internet (ICMP).

```
!--- This command is used to permit Telnet traffic
!--- from machine 10.1.1.2 to machine 172.16.1.1. access-list 101 permit tcp host 10.1.1.2 host
172.16.1.1 eq telnet


!--- This command is used to permit tcp traffic from
!--- 10.1.1.2 host machine to 172.16.1.1 host machine. access-list 101 permit tcp host 10.1.1.2
host 172.16.1.1


!--- This command is used to permit udp traffic from
```

```
!--- 10.1.1.2 host machine to 172.16.1.1 host machine. access-list 101 permit udp host 10.1.1.2
host 172.16.1.1


!--- This command is used to permit ip traffic from
!--- 10.1.1.0 network to 172.16.1.10 network. access-list 101 permit ip 10.1.1.0 0.0.0.255
172.16.1.0 0.0.0.255
```

Definición de los Puertos y los Tipos de Mensaje

No sólo puede definir el origen y el destino de ACL, sino que también puede definir puertos, tipos de mensajes ICMP y otros parámetros. Una buena fuente de información para los puertos conocidos es RFC 1700. Los tipos de mensaje de ICMP se explican en RFC 792.

El router puede mostrar texto descriptivo en algunos de los puertos conocidos. Use un **?**para obtener ayuda.

```
access-list 102 permit tcp host 10.1.1.1 host 172.16.1.1 eq ?
  bgp          Border Gateway Protocol (179)
  chargen      Character generator (19)
  cmd          Remote commands (rcmd, 514)
```

Durante la configuración, el router también convierte valores numéricos a valores más fáciles de utilizar. Este es un ejemplo donde escribe el número de tipo de mensaje ICMP y hace que el router convierta el número en un nombre.

```
access-list 102 permit icmp host 10.1.1.1 host 172.16.1.1 14
```

se convierte en

```
access-list 102 permit icmp host 10.1.1.1 host 172.16.1.1 timestamp-reply
```

Aplicación de ACL

Puede 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. Como se muestra en este ejemplo, cuando intenta bloquear el tráfico de origen a destino, puede aplicar una ACL entrante a E0 en el router A en lugar de una lista saliente a E1 en el router C. Una lista de acceso tiene **adeny ip 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 255.255.2 55. El puerto de origen es 68 y el de destino 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.

> **Nota:** Para que el tráfico UDP pase, la ACL también debe permitir explícitamente el tráfico UDP.

Definición de Entrada, Salida, Entrante, Saliente, Origen y Destino

El router utiliza los términos entrada, salida, origen y destino como referencias. El tráfico en el router se puede comparar con el tráfico en una carretera. Si usted era un agente de la ley en Pennsylvania y quería detener un camión que viaja de Maryland a Nueva York, la fuente del camión es Maryland, y el destino del camión es Nueva York. La barricada podría aplicarse en la frontera entre Pensilvania y Nueva York (fuera) o en la frontera entre Maryland y Pensilvania (dentro).

Cuando usted se refiere a un router, estos términos tienen estos significados.

- **Salida: tráfico que ya ha pasado a través del router y sale de la interfaz.** El origen es el lugar en donde ha estado, en el otro lado del router, y el destino es el lugar a donde va.
- **Entrada: tráfico que llega a la interfaz y después pasa a través del router.** El origen es el lugar en donde ha estado y el destino es el lugar a donde va, en el otro lado del router.
- **Entrante: si la lista de acceso es entrante, cuando el router recibe un paquete, el Cisco IOS Software verifica las declaraciones de criterios de la lista de acceso para obtener una coincidencia.** Si se permite el paquete, el software continúa procesando el paquete. Si se niega el paquete, el software descarta el paquete.
- **Saliente: si la lista de acceso es saliente, una vez que el software haya recibido y ruteado un paquete hacia la interfaz saliente, el software verificará las declaraciones de criterios de la lista de acceso para obtener una coincidencia.** Si se permite el paquete, el software transmite el paquete. Si se niega el paquete, el software descarta el paquete.

La ACL de entrada tiene un origen en un segmento de la interfaz a la que se aplica y un destino fuera de cualquier otra interfaz. La ACL de salida tiene un origen en un segmento de cualquier interfaz diferente a la interfaz a la que se aplica y un destino fuera de la interfaz a la que se aplica.

Edición de ACL

La edición de una ACL requiere especial atención. Por ejemplo, si usted intenta eliminar una línea específica de una ACL numerada existente como se muestra aquí, se eliminará la ACL entera.

```
!--- The access-list 101 denies icmp from any to any network
!--- but permits IP traffic from any to any network. router#configure terminal
  Enter configuration commands, one per line.  End with CNTL/Z.
  router(config)#access-list 101 deny icmp any any
  router(config)#access-list 101 permit ip any any
  router(config)#^Z

  router#show access-list
  Extended IP access list 101
      deny icmp any any
      permit ip any any
  router#
  *Mar  9 00:43:12.784: %SYS-5-CONFIG_I: Configured from console by console
```

```
router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router(config)#no access-list 101 deny icmp any any
router(config)#^Z

router#show access-list
router#
*Mar  9 00:43:29.832: %SYS-5-CONFIG_I: Configured from console by console
```

Copie la configuración del router en un servidor TFTP o en un editor de textos como Bloc de notas para editar ACL numeradas. Después, realice los cambios necesarios y copie la configuración nuevamente en el router.

También puede hacer esto.

```
router#configure terminal
  Enter configuration commands, one per line.
  router(config)#ip access-list extended test

!--- Permits IP traffic from 10.2.2.2 host machine to 10.3.3.3 host machine. router(config-ext-
nacl)#permit ip host 10.2.2.2 host 10.3.3.3

!--- Permits www traffic from 10.1.1.1 host machine to 10.5.5.5 host machine. router(config-ext-
nacl)#permit tcp host 10.1.1.1 host 10.5.5.5 eq www

!--- Permits icmp traffic from any to any network. router(config-ext-nacl)#permit icmp any any

!--- Permits dns traffic from 10.6.6.6 host machine to 10.10.10.0 network. router(config-ext-
nacl)#permit udp host 10.6.6.6 10.10.10.0 0.0.0.255 eq domain
  router(config-ext-nacl)#^Z
  1d00h: %SYS-5-CONFIG_I: Configured from console by consoles-l

  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
      permit udp host 10.6.6.6 10.10.10.0 0.0.0.255 eq domain
```

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

```
router#configure terminal
   Enter configuration commands, one per line.  End with CNTL/Z.
   router(config)#ip access-list extended test

!--- ACL entry deleted. router(config-ext-nacl)#no permit icmp any any

!--- ACL entry added. router(config-ext-nacl)#permit gre host 10.4.4.4 host 10.8.8.8
   router(config-ext-nacl)#^Z
   1d00h: %SYS-5-CONFIG_I: Configured from console by consoles-l

   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 gre host 10.4.4.4 host 10.8.8.8
```

También puede agregar líneas de ACL a ACL extendidas numeradas o estándar numeradas por el número de secuencia en el Cisco IOS. Este es un ejemplo de la configuración:

Configure la ACL extendida de esta manera:

```
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
Router(config)#exit
Router#
```

Ejecute el comando **show access-**list para ver las entradas de ACL. Los números de secuencia como 10, 20 y 30 también aparecen aquí.

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

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

Ejemplo 1:

```
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip access-list extended 101
Router(config-ext-nacl)#5 deny tcp any any eq telnet
Router(config-ext-nacl)#exit
Router(config)#exit
Router#
```

En el resultado del comando **show access-**list, la ACL del número de secuencia 5 se agrega como la primera entrada a la lista de acceso 101.

```
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
Router#
```

Ejemplo 2:

```
internetrouter#show access-lists
Extended IP access list 101
    10 permit tcp any any
    15 permit tcp any host 172.16.2.9
    20 permit udp host 172.16.1.21 any
    30 permit udp host 172.16.1.22 any

internetrouter#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
internetrouter(config)#ip access-list extended 101
internetrouter(config-ext-nacl)#18 per tcp any host 172.16.2.11
internetrouter(config-ext-nacl)#^Z

internetrouter#show access-lists
Extended IP access list 101
    10 permit tcp any any
```

```
    15 permit tcp any host 172.16.2.9
    18 permit tcp any host 172.16.2.11
    20 permit udp host 172.16.1.21 any
    30 permit udp host 172.16.1.22 any
internetrouter#
```

De forma similar, puede configurar la lista de acceso estándar de esta manera:

```
internetrouter(config)#access-list 2 permit 172.16.1.2
internetrouter(config)#access-list 2 permit 172.16.1.10
internetrouter(config)#access-list 2 permit 172.16.1.11

internetrouter#show access-lists
Standard IP access list 2
    30 permit 172.16.1.11
    20 permit 172.16.1.10
    10 permit 172.16.1.2

internetrouter(config)#ip access-list standard 2
internetrouter(config-std-nacl)#25 per 172.16.1.7
internetrouter(config-std-nacl)#15 per 172.16.1.16

internetrouter#show access-lists
Standard IP access list 2
    15 permit 172.16.1.16
    30 permit 172.16.1.11
    20 permit 172.16.1.10
    25 permit 172.16.1.7
    10 permit 172.16.1.2
```

La diferencia principal en una lista de acceso estándar es que el IOS de Cisco agrega una entrada en orden descendente de la dirección IP, no en un número de secuencia.

En este ejemplo, se muestran las diferentes entradas; por ejemplo, cómo permitir una dirección IP (192.168.100.0) o las redes (10.10.10.0).

```
internetrouter#show access-lists
Standard IP access list 19
    10 permit 192.168.100.0
    15 permit 10.10.10.0, wildcard bits 0.0.0.255
    19 permit 10.101.110.0, wildcard bits 0.0.0.255
    25 deny any
```

Agregue la entrada en la lista de acceso 2 para permitir la dirección IP 172.22.1.1:

```
internetrouter(config)#ip access-list standard 2
internetrouter(config-std-nacl)#18 permit 172.22.1.1
```

Esta entrada se agrega en la parte superior de la lista para dar prioridad a la dirección IP específica, en lugar de la red.

```
internetrouter#show access-lists
Standard IP access list 19
    10 permit 192.168.100.0
    18 permit 172.22.1.1
    15 permit 10.10.10.0, wildcard bits 0.0.0.255
    19 permit 10.101.110.0, wildcard bits 0.0.0.255
    25 deny   any
```

Nota: Las ACL anteriores no se soportan en un dispositivo de seguridad como ASA/PIX Firewall.

Pautas para cambiar las listas de acceso cuando se aplican a mapas crypto

- Si agrega a una configuración de lista de acceso actual, no es necesario quitar el mapa criptográfico. Si usted agrega a esta directamente sin la remoción del mapa crypto, esto se soporta y es aceptable.
- Si necesita modificar o eliminar una entrada de la lista de acceso de una lista de acceso actual, debe eliminar el mapa criptográfico de la interfaz. Una vez que haya quitado el mapa crypto, realice todos los cambios necesarios a la lista de acceso y vuelva a agregar el mapa crypto. Si realiza cambios como la eliminación de la lista de acceso sin la remoción del mapa crypto, esto no se soporta y puede dar lugar a un comportamiento impredecible.

Troubleshoot

¿Cómo elimino una ACL de una interfaz?

Ingrese al modo de configuración y escriba **no delante del comando access-group, como se muestra en este ejemplo, para quitar una ACL de una interfaz.**

```
interface <interface-name> no ip access-group <acl-number> {in|out}
```

¿Qué hago cuando se niega demasiado tráfico?

Si se niega demasiado tráfico, estudie la lógica de su lista o intente definir y aplicar otra lista más amplia. El **comando show ip access-lists proporciona un conteo de paquetes que muestra qué entrada de ACL se consulta.** La palabra clave log al final de las entradas de ACL individuales muestra el número de ACL y si el paquete fue permitido o negado, además de información específica de puerto.

Nota: La palabra clave log-input existe en el Cisco IOS Software, versión 11.2 y posteriores, y en el Cisco IOS Software, versión 11.1 basado en software creado específicamente para el mercado de proveedores de servicios. El software anterior no admite esta palabra clave. El uso de esta palabra clave incluye la interfaz de entrada y la dirección MAC de origen, donde corresponda.

¿Cómo ejecuto un debug en el nivel de paquete que utiliza un router de Cisco?

Este procedimiento explica el proceso de debug. Antes de comenzar, asegúrese de que no haya ACL actualmente aplicadas, de que haya una ACL y de que el fast switching no esté inhabilitado.

Nota: Tenga mucho cuidado cuando ejecute un debug de un sistema con mucho tráfico. Utilice una ACL para ejecutar un debug de tráfico específico. Pero asegúrese del proceso y del flujo de tráfico.

1. Utilice **el comando access-**list para capturar los datos deseados.En este ejemplo, la captura

de datos está configurada para la dirección de destino de 10.2.6.6 o la dirección de origen de 10.2.6.6.

```
access-list 101 permit ip any host 10.2.6.6
access-list 101 permit ip host 10.2.6.6 any
```

2. Inhabilite el fast switching en las interfaces involucradas. Solo verá el primer paquete si el fast switching no está inhabilitado.

```
configure terminal
interface
```

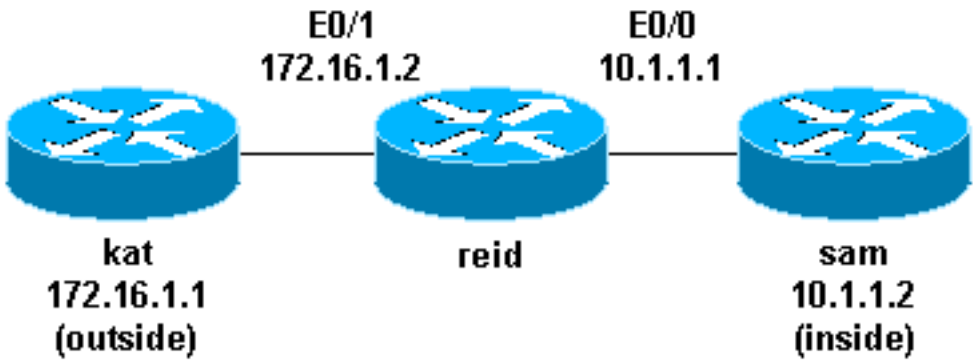
3. Utilice el **comando terminal monitor en el modo de habilitación para mostrar el resultado del comando debug y los mensajes de error del sistema para la sesión y el terminal actuales.**
4. Utilice el comando **debug ip packet 101** o el comando **debug ip packet 101 detail** para comenzar el proceso de depuración.
5. Ejecute el **comando no debug all en el modo de habilitación y el comando interface configuration para detener el proceso de debug.**
6. Vuelva a iniciar el almacenamiento en memoria caché.

```
configure terminal
interface
```

Tipos de ACL IP

Esta sección del documento describe tipos de ACL.

Diagrama de la red



ACL Estándar

Las ACL estándar son el tipo más antiguo de ACL. Se remontan a la versión 8.3 del software Cisco IOS. Las ACL estándar controlan el tráfico mediante la comparación de la dirección de origen de los paquetes IP con las direcciones configuradas en la ACL.

Este es el formato de sintaxis del comando de una ACL estándar.

```
access-list <access-list-number> {permit|deny} {host|source source-wildcard|any}
```

En todas las versiones de software, el *access-list-number* puede ser cualquier valor entre 1 y 99.

En la versión 12.0.1 del software del IOS de Cisco, las ACL estándar comienzan a utilizar números adicionales (1300 a 1999). Estos números adicionales se denominan ACL IP extendidas. El Cisco IOS Software, versión 11.2 agregó la capacidad para utilizar el nombre de lista en las ACL estándar.

Una configuración *source/source-wildcard* de 0.0.0.0/255.255.255.255 se puede especificar como **any** . El comodín puede omitirse si tiene todos ceros. Por lo tanto, el host 10.1.1.2 0.0.0.0 es el mismo que el host 10.1.1.2.

Después de definir la ACL, debe aplicarse a la interfaz (entrante o saliente). En las primeras versiones de software, out era el valor predeterminado cuando no se especificaba una palabra clave out o in. La dirección debe ser especificada en versiones de software posteriores.

```
interface <interface-name>
 ip access-group number {in|out}
```

Este es un ejemplo del uso de una ACL estándar para bloquear todo el tráfico, excepto el tráfico con origen en 10.1.1.x.

```
interface Ethernet0/0
 ip address 10.1.1.1 255.255.255.0
 ip access-group 1 in
!
access-list 1 permit 10.1.1.0 0.0.0.255
```

ACL Extendidas

Las ACL extendidas se introdujeron en Cisco IOS Software Release 8.3. Las ACL extendidas controlan el tráfico mediante la comparación de las direcciones de origen y destino de los paquetes IP con las direcciones configuradas en la ACL.

Este es el formato de sintaxis del comando de las ACL extendidas. Las líneas se ajustan aquí por consideraciones de espacio.

IP

```
access-list access-list-number
     [dynamic dynamic-name [timeout minutes]]
     {deny|permit} protocol source source-wildcard destination destination-wildcard [precedence
precedence]
     [tos tos] [log|log-input] [time-range time-range-name]
```

ICMP

```
access-list access-list-number
     [dynamic dynamic-name [timeout minutes]]
     {deny|permit} icmp source source-wildcard destination destination-wildcard
     [icmp-type [icmp-code] |icmp-message] [precedence precedence] [tos tos] [log|log-input]
     [time-range time-range-name]
```

TCP

```
access-list access-list-number
     [dynamic dynamic-name [timeout minutes]]
     {deny|permit} tcp source source-wildcard [operator [port]]
     destination destination-wildcard [operator [port]]
     [established] [precedence precedence] [tos tos]
     [log|log-input] [time-range time-range-name]
```

UDP

```
access-list access-list-number
     [dynamic dynamic-name [timeout minutes]]
     {deny|permit} udp source source-wildcard [operator [port]]
     destination destination-wildcard [operator [port]]
     [precedence precedence] [tos tos] [log|log-input]
     [time-range time-range-name]
```

En todas las versiones de software, el *access-list-number* puede ser de 100 a 199. En Cisco IOS Software Release 12.0.1, las ACL extendidas comienzan a utilizar números adicionales (2000 a 2699). Estos números adicionales se denominan ACL IP extendidas. El Cisco IOS Software, versión 11.2 agregó la capacidad para utilizar el nombre de lista en las ACL extendidas.

El valor de 0.0.0.0/255.255.255.255 se puede especificar como **cualquiera.** Después de definir la ACL, debe aplicarse a la interfaz (entrante o saliente). En las primeras versiones de software, out era el valor predeterminado cuando no se especificaba una palabra clave out o in. La dirección debe ser especificada en versiones de software posteriores.

```
interface <interface-name>
 ip access-group {number|name} {in|out}
```

Esta ACL extendida se utiliza para permitir el tráfico en la red 10.1.1.x (interna) y para recibir respuestas de ping del exterior, mientras evita los pings no solicitados de personas externas, lo que permite todo el otro tráfico.

```
interface Ethernet0/1
 ip address 172.16.1.2 255.255.255.0
 ip access-group 101 in
!
access-list 101 deny icmp any 10.1.1.0 0.0.0.255 echo access-list 101 permit ip any 10.1.1.0
0.0.0.255
```

> **Nota:** Algunas aplicaciones como la administración de la red requieren pings para una función de keepalive. Si este es el caso, puede limitar los pings entrantes que están bloqueados o ser más granulares en las IP permitidas/denegadas.

Lock and Key (ACL dinámicas)

Lock and key, también conocido como ACL dinámicas, se introdujo en Cisco IOS Software Release 11.1. Esta función depende de Telnet, autenticación (local o remota) y ACL extendidas.

La configuración de cerradura y llave comienza con la aplicación de una ACL extendida para bloquear el tráfico a través del router. Los usuarios que desean atravesar el router son bloqueados por la ACL extendida hasta que realicen una conexión Telnet al router y sean autenticados. La conexión Telnet se interrumpe y se agrega una ACL dinámica de entrada única a la ACL extendida que existe. Esto permite el tráfico por un período de tiempo determinado; son posibles los tiempos de espera inactivo y absoluto.

Este es el formato de sintaxis del comando para la configuración de cerradura y llave con autenticación local.

```
username <user-name> password <password>
!
interface <interface-name>
 ip access-group {number|name} {in|out}
```

La ACL de una única entrada en este comando se agrega dinámicamente a la ACL existente después de la autenticación.

```
access-list access-list-number dynamic name {permit|deny} [protocol]
{source source-wildcard|any} {destination destination-wildcard|any}
[precedence precedence][tos tos][established] [log|log-input]
[operator destination-port|destination port]

line vty <line_range>
login local
```

Este es un ejemplo básico de cerradura y llave.

```
username test password 0 test

!--- Ten (minutes) is the idle timeout. username test autocommand access-enable host timeout 10
!
interface Ethernet0/0
  ip address 10.1.1.1 255.255.255.0
  ip access-group 101 in
!
access-list 101 permit tcp any host 10.1.1.1 eq telnet

!--- 15 (minutes) is the absolute timeout. access-list 101 dynamic testlist timeout 15 permit ip
10.1.1.0 0.0.0.255 172.16.1.0 0.0.0.255
!
line vty 0 4
 login local
```

Una vez que el usuario en 10.1.1.2 haya realizado una conexión Telnet a 10.1.1.1, se aplicará la ACL dinámica. Luego, se pierde la conexión y el usuario puede ir a la red 172.16.1.x.

ACL con nombre IP

Las ACL con nombre IP se introdujeron en Cisco IOS Software Release 11.2. Esto permite que a las ACL estándar y extendidas se les asignen nombres en lugar de números.

Este es el formato de sintaxis del comando de las ACL con nombre IP.

```
ip access-list {extended|standard} name
```

Este es un ejemplo de TCP:

```
{permit|deny} tcp source source-wildcard [operator [port]] destination destination-wildcard
[operator [port]] [established] [precedence precedence] [tos tos] [log] [time-range time-range-
name]
```

Este es un ejemplo del uso de una ACL con nombre para bloquear todo el tráfico, excepto la conexión Telnet del host 10.1.1.2 al host 172.16.1.1.

```
interface Ethernet0/0
 ip address 10.1.1.1 255.255.255.0
 ip access-group in_to_out in
!
ip access-list extended in_to_out
 permit tcp host 10.1.1.2 host 172.16.1.1 eq telnet
```

ACL Reflexivas

Las ACL reflexivas se introdujeron en Cisco IOS Software Release 11.3. Las ACL reflexivas permiten que los paquetes IP se filtren según la información de sesión de la capa superior. Generalmente, se utilizan para permitir el tráfico saliente y para limitar el tráfico entrante en respuesta a las sesiones que se originan dentro del router.

Las ACL reflexivas solo se pueden definir con ACL con nombre IP extendidas. No se pueden definir con ACL con nombre IP estándar ni numeradas, ni con otras ACL de protocolo. Las ACL reflexivas pueden utilizarse conjuntamente con otras ACL ampliadas estándar y estáticas.

Esta es la sintaxis de varios comandos de ACL reflexivas.

```
interface <interface-name>
 ip access-group {number|name} {in|out}
!
ip access-list extended <name>
 permit protocol any any reflect name [timeoutseconds]
!
ip access-list extended <name>
 evaluate <name>
```

Este es un ejemplo del permiso de tráfico entrante y saliente ICMP, mientras que permite solamente el tráfico TCP que se ha iniciado desde adentro, el otro tráfico se niega.

```
ip reflexive-list timeout 120
!
interface Ethernet0/1
 ip address 172.16.1.2 255.255.255.0
 ip access-group inboundfilters in
 ip access-group outboundfilters out
!
ip access-list extended inboundfilters
```

```
 permit icmp 172.16.1.0 0.0.0.255 10.1.1.0 0.0.0.255
 evaluate tcptraffic

!--- This ties the reflexive ACL part of the outboundfilters ACL,
!--- called tcptraffic, to the inboundfilters ACL. ip access-list extended outboundfilters
 permit icmp 10.1.1.0 0.0.0.255 172.16.1.0 0.0.0.255
 permit tcp 10.1.1.0 0.0.0.255 172.16.1.0 0.0.0.255 reflect tcptraffic
```

ACL Basadas en Tiempo que Utilizan Intervalos de Tiempo

Las ACL basadas en tiempo se introdujeron en el Cisco IOS Software, versión 12.0.1.T. Aunque su función es similar a la de los ACL extendidos, permiten el control de acceso en base al tiempo. Se crea un intervalo de tiempo que define las horas específicas del día y de la semana para implementar las ACL basadas en tiempo. El intervalo de tiempo se identifica con un nombre y luego se remite a él a través de una función. Por lo tanto, las restricciones de tiempo se imponen en la misma función. El intervalo de tiempo depende del reloj del sistema del router. Se puede utilizar el reloj del router, pero la función funciona mejor con la sincronización de Network Time Protocol (NTP).

Estos son los comandos de ACL basadas en tiempo.

```
!--- Defines a named time range. time-range time-range-name

!--- Defines the periodic times. periodic days-of-the-week hh:mm to [days-of-the-week] hh:mm

!--- Or, defines the absolute times. absolute [start time date] [end time date]

!--- The time range used in the actual ACL. ip access-list name|number time-rangename_of_time-
range
```

En este ejemplo, se permite una conexión Telnet de la red interna a la red externa el lunes, el miércoles y el viernes durante el horario comercial:

```
interface Ethernet0/0
 ip address 10.1.1.1 255.255.255.0
 ip access-group 101 in
!
access-list 101 permit tcp 10.1.1.0 0.0.0.255 172.16.1.0 0.0.0.255 eq telnet time-range
EVERYOTHERDAY
!
time-range EVERYOTHERDAY
 periodic Monday Wednesday Friday 8:00 to 17:00
```

Entradas de ACL IP Comentadas

Las entradas de ACL IP comentadas se introdujeron en el Cisco IOS Software, versión 12.0.2.T. Los comentarios facilitan la comprensión de las ACL y se pueden utilizar para ACL IP extendidas o estándar.

Esta es la sintaxis del comando de las ACL con nombre IP comentadas.

```
ip access-list {standard|extended} <access-list-name> remark remark
```

Esta es la sintaxis del comando de las ACL IP numeradas comentadas.

```
access-list <access-list-number> remark remark
```

Este es un ejemplo de comentarios dentro de una ACL numerada.

```
interface Ethernet0/0
 ip address 10.1.1.1 255.255.255.0
 ip access-group 101 in
!
access-list 101 remark permit_telnet
access-list 101 permit tcp host 10.1.1.2 host 172.16.1.1 eq telnet
```

Control de Acceso Basado en Contexto

El control de acceso basado en contexto (CBAC) se introdujo en el Cisco IOS Software, versión 12.0.5.T, y requiere el conjunto de funciones del Cisco IOS Firewall. El CBAC examina el tráfico que viaja a través del firewall para detectar y manejar la información de estado para las sesiones de UDP y TCP. Esta información de estado se utiliza para crear las aperturas temporales en las listas de acceso del firewall. Configure listas de inspección IP en la dirección del flujo de iniciación del tráfico para permitir el tráfico de retorno y conexiones de datos adicionales para la sesión permitida, sesiones que se originaron dentro de la red interna protegida, para hacer esto.

Esta es la sintaxis para CBAC.

```
ip inspect name inspection-name protocol [timeoutseconds]
```

Este es un ejemplo del uso de CBAC para examinar el tráfico saliente. La ACL extendida 111 normalmente bloqueará el tráfico de retorno que no sea de ICMP sin agujeros de apertura de CBAC para el tráfico de retorno.

```
ip inspect name myfw ftp timeout 3600
ip inspect name myfw http timeout 3600
ip inspect name myfw tcp timeout 3600
ip inspect name myfw udp timeout 3600
ip inspect name myfw tftp timeout 3600
! interface Ethernet0/1 ip address 172.16.1.2 255.255.255.0 ip access-group 111 in ip inspect
myfw out !
access-list 111 deny icmp any 10.1.1.0 0.0.0.255 echo access-list 111 permit icmp any 10.1.1.0
0.0.0.255
```

Proxy de Autenticación

El proxy de autenticación se introdujo en el Cisco IOS Software, versión 12.0.5.T. Esto requiere que usted tenga el conjunto de funciones del Cisco IOS Firewall. El servidor alterno de autenticación se utiliza para autenticar usuarios de entrada, de salida o ambos. Los usuarios que son bloqueados normalmente por una ACL pueden hacer uso de un navegador para atravesar el firewall y realizar la autenticación en un servidor de RADIUS o TACACS+. El servidor transmite entradas de ACL adicionales al router para permitir que los usuarios pasen después de la autenticación.

El servidor alterno de autenticación es similar a Lock and Key (ACL dinámicas). Estas son las diferencias:

- La función de cerradura y llave es activada por una conexión Telnet al router. El proxy de autenticación es activado por HTTP a través del router.
- El proxy de autenticación debe utilizar un servidor externo.
- El proxy de autenticación puede manejar la adición de varias listas dinámicas. La función de cerradura y llave puede agregar solo una.
- El proxy de autenticación tiene un tiempo de espera absoluto, pero ningún tiempo de espera inactivo. La función de cerradura y llave tiene ambos.

Consulte Compendio de Configuraciones de Software Seguras e Integradas de Cisco para ver ejemplos de proxy de autenticación.

Turbo ACL

Las Turbo ACL se introdujeron en el Cisco IOS Software, versión 12.1.5.T, y solo se encuentran en las plataformas 7200, 7500 y otras de alta capacidad. La función de turbo ACL se diseñó para procesar las ACL más eficazmente a fin de mejorar el rendimiento del router.

Utilice el **comando access-list compiled para las turbo ACL.** Este es un ejemplo de una ACL compilada.

```
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 ftp
access-list 101 permit udp host 10.1.1.2 host 172.16.1.1 eq syslog
access-list 101 permit udp host 10.1.1.2 host 172.16.1.1 eq tftp
access-list 101 permit udp host 10.1.1.2 host 172.16.1.1 eq ntp
```

Después de definir la ACL extendida o estándar, utilice el **comando global configuration para compilarla.**

```
!--- Tells the router to compile. access-list compiled
!
interface Ethernet0/1
 ip address 172.16.1.2 255.255.255.0

!--- Applies to the interface.  ip access-group 101 in
```

El **comando show access-list compiled muestra estadísticas sobre las ACL.**

ACL Distribuidas Basadas en Tiempo

Las ACL distribuidas basadas en tiempo se introdujeron en el Cisco IOS Software, versión 12.2.2.T para implementar las ACL basadas en tiempo en los 7500 Series Routers con VPN habilitada. Antes de la introducción de la función de ACL distribuida basada en tiempo, las ACL basadas en tiempo no eran soportadas en linecards para los Cisco 7500 Series Routers. Si se configuraban ACL basadas en tiempo, estas funcionaban como ACL normales. Si una interfaz en una linecard se configuraba con ACL basadas en tiempo, los paquetes conmutados en la interfaz no se distribuían conmutados a través de la linecard, sino que se reenviaban al procesador de ruta para su proceso.

La sintaxis para las ACL distribuidas basadas en tiempo es la misma que para las ACL basadas

en tiempo con la adición de los comandos con respecto al estado de los mensajes de comunicación entre procesadores (IPC) entre el procesador de ruta y la tarjeta de línea.

```
debug time-range ipc
show time-range ipc
clear time-range ipc
```

ACL de recepción

Las ACL recibidas se utilizan para aumentar la seguridad en los Cisco 12000 Routers a través de la protección del procesador de ruta gigabit (GRP) del router contra el tráfico innecesario y potencialmente nefario. Las ACL recibidas se agregaron como exención especial a la válvula reguladora de mantenimiento para el Cisco IOS Software, versión 12.0.21S2, y se integraron en 12.0(22)S. Consulte GSR: Recibir listas de control de acceso para obtener más información.

ACL de Protección de Infraestructura

Las ACL de infraestructura se utilizan para minimizar el riesgo y la eficacia de un ataque directo a la infraestructura mediante el permiso explícito de solo el tráfico autorizado al equipo de infraestructura, mientras que permite el resto del tráfico de tránsito. Consulte Protección del Núcleo: Consulte Listas de Control de Acceso de Protección de Infraestructura para obtener más información.

ACL de Tránsito

Las ACL de tránsito se utilizan para aumentar la seguridad de la red, ya que permiten explícitamente solo tráfico requerido en sus redes. Consulte Listas de Control de Acceso de Tránsito: Consulte Filtración en su Borde para obtener más información.

Información Relacionada

- Configurar ACL de IP de uso general
- RFC 1700
- RFC 1918
- Páginas de Soporte de Listas de Acceso
- Cisco IOS Firewall
- Soporte Técnico y Documentación - Cisco Systems

Acerca de esta traducción
Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).