Configuración de listas de acceso IP

Contenidos

Introducción

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

Acceda a la herramienta asesor de software (sólo para clientes registrados) para determinar la compatibilidad de algunas de las características avanzadas de las ACL IP de Cisco IOS®.

RFC 1700 contiene los números asignados de los puertos conocidos. RFC 1918 contiene la asignación de direcciones de las redes internas privadas (direcciones IP que no deberían verse en Internet).

Nota: las ACL también podrían emplearse para fines adicionales al filtrado del tráfico IP, por ejemplo, para definir el tráfico a la traducción de direcciones de red (NAT) o el cifrado, o para filtrar protocolos distintos de IP tales como AppleTalk o IPX. Una discusión de estas funciones está fuera del alcance de este documento.

Requisitos previos

Requisitos

No hay requisitos previos específicos para este documento. Los conceptos explicados se encuentran en las versiones 8.3 o posteriores del software Cisco IOS®. Esto se indica a continuación de cada función de lista de acceso.

Componentes utilizados

En este documento se describen diversos tipos de ACL. Algunas ya estaban presentes en las versiones 8.3 del software Cisco IOS, y otras se incluyeron en versiones posteriores. Este hecho se indica en el comentario de cada tipo.

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración despejada (predeterminada). Si la red está en producción, asegúrese de comprender el impacto que pueda tener cualquier comando.

Convenciones

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

Conceptos de ACL

Esta sección describe los conceptos de ACL.

Máscaras

Las máscaras se usan con direcciones IP en ACL IP para especificar qué debe permitirse y qué debe denegarse. Las máscaras para configurar direcciones IP en interfaces empiezan por 255 y tienen los valores más altos a la izquierda (por ejemplo: dirección IP 209.165.202.129 con una máscara de 255.255.255.224). Las máscaras para las ACL IP son lo opuesto, por ejemplo, máscara 0.0.0.255. Esto a veces se denomina una máscara inversa o una máscara comodín. Cuando el valor de la máscara se subdivide en binario (0 y 1), el resultado determina qué bits de dirección se considerarán en el procesamiento del tráfico. La presencia de un 0 indica que deben tenerse en cuenta los bits de la dirección (coincidencia exacta); un 1 en la máscara indica que no es relevante. En esta tabla se amplía el concepto.

Ejemplo de máscara	
dirección de red (tráfico que se va a procesar)	*10.1.1.0*
máscara	*0.0.0.255*
dirección de red (binaria)	*00001010.00000001.00000001.00000000*
máscara (binaria)	*00000000.00000000.00000000.11111111*

Basándose en la máscara binaria, puede ver que los primeros tres grupos (octetos) deben coincidir exactamente con la dirección de red binaria dada (00001010.00000001.00000001). Los últimos conjuntos de números no son relevantes (.11111111). Por lo tanto, todo el tráfico que empiece por 10.1.1. coincidiría, puesto que el último octeto no es relevante. Por consiguiente, con esta máscara, se procesarán las direcciones desde la 10.1.1.1 hasta la 10.1.1.255 (10.1.1.x).

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

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

Tenga en cuenta estos equivalentes de ACL.

- El valor source/source-wildcard de 0.0.0.0/255.255.255.255 significa cualquiera.
- El valor source/wildcard de 10.1.1.2/0.0.0.0 es el mismo que 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. Tenemos 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 se explica cómo resumirlas en una sola.

El tercer octeto de las redes anteriores se puede escribir como se indica en esta tabla, según la posición del bit del octeto y el valor de dirección de 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

Como los cinco primeros 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 bajo son relevantes para los rangos de red en cuestión. Este comando define una ACL que permite esta red. Si resta 255.255.248.0 (máscaa normal) de 255.255.255.255, el resultado es 0.0.7.255.

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

Tenga en cuenta este conjunto de redes para profundizar en el tema.

```
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 se explica cómo resumirlas.

El tercer octeto de las redes anteriores se puede escribir como se indica en esta tabla, según la posición del bit del octeto y el valor de dirección de 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, estas redes no se pueden resumir en una sola, se necesitan dos como mínimo. Las redes anteriores pueden resumirse en estas dos redes:

- En el caso de las redes 192.168.146.x y 192.168.147.x, coinciden todos los bits menos el último, que no es relevante. Esto puede escribirse como 192.168.146.0/23 (o 192.168.146.0 255.255.254.0).
- En el caso de las redes 192.168.148.x y 192.168.149.x, también coinciden todos los bits menos el último, que nuevamente no es relevante. Esto puede escribirse como 192.168.148.0/23 (o 192.168.148.0 255.255.254.0).

El siguiente resultado define la ACL resumida de las redes anteriores.

```
          access-list 10 permit ip 192.168.146.0 0.0.1.255
access-list 10 permit ip 192.168.148.0 0.0.1.255
```

Procesamiento de ACL

El tráfico que entra en el router se compara con las entradas de ACL según el orden de las entradas en el router. Se agregan nuevas sentencias al final de la lista. El router sigue mirando hasta que encuentra una coincidencia. Si el router llega al final de la lista y no ha encontrado ninguna coincidencia, el tráfico se rechaza. Por este motivo, debe tener las entradas consultadas con frecuencia al principio de la lista. Hay un rechazo implícito para el tráfico que no está permitido. Una ACL de única entrada con una sola entrada "deny" tiene el efecto de rechazar todo el tráfico. Si no tiene como mínimo una sentencia "permit" en una ACL, se bloqueará todo el tráfico. Estas dos ACL (101 y 102) tienen el mismo efecto.

```
          access-list 101 permit ip 10.1.1.0 0.0.0.255 172.16.1.0 0.0.0.255

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 este ejemplo, la última entrada es suficiente. No necesita las tres primeras entradas porque TCP incluye Telnet e IP incluye TCP, el Protocolo de datagrama de usuario (UDP) y el Protocolo de mensajes de control de Internet (ICMP).

```
          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
access-list 101 permit udp host 10.1.1.2 host 172.16.1.1
access-list 101 permit ip 10.1.1.0 0.0.0.255 172.16.1.0 0.0.0.255
```

Definición de puertos y tipos de mensajes

Además de definir el origen y el destino de las ACL, es posible definir los puertos, los tipos de mensajes ICMP y otros parámetros. Una buena fuente de información para los puertos conocidos es RFC 1700 . Los tipos de mensajes ICMP se explican en RFC 792 .

El router puede mostrar textos descriptivos de algunos puertos conocidos. Use un símbolo **?** 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 en valores más fáciles de utilizar. En este ejemplo, al escribir el número de tipo de mensaje ICMP, el router convierte 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

Es posible definir las ACL sin aplicarlas. No obstante, las ACL no surten efecto hasta que se aplican a la interfaz del router. Por tanto, se recomienda aplicar la ACL en la interfaz más próxima al origen del tráfico. Como se muestra en este ejemplo, cuando intente bloquear el tráfico del origen al destino, puede aplicar una ACL entrante a E0 en el router A en vez de una lista saliente a E1 en el router C.

Definición de In, Out, Source y Destination

El router utiliza los términos "in", "out", "source" y "destination" como referencias. Se podría comparar el tráfico del router con el tráfico de una autopista. Si fuera un agente de policía de Lérida y quisiera parar un camión que va de Tarragona a Barcelona, el origen del camión sería Tarragona y su destino Barcelona. El control de carretera se colocaría en la frontera entre Lérida y Barcelona ("out") o en la frontera entre Tarragona y Lérida ("in").

Aplicados a un router, dichos términos tienen los siguientes significados.

- **Out** : el tráfico que ya ha pasado por el router y está saliendo de la interfaz. El origen es por donde ha pasado (en el otro extremo del router) y el destino es adonde va.
- **In**: el tráfico que llega a la interfaz y luego pasa por el router. El origen es por donde ha pasado y el destino es adonde va (en el otro extremo del router).

La ACL "in" tiene un origen en un segmento de la interfaz al que se aplica y un destino fuera de cualquier otra interfaz. La ACL "out" tiene un origen en un segmento de cualquier interfaz distinta a la interfaz a la que se aplica y un destino fuera de la interfaz a la que se aplica.

Edición de ACL

La modificación de una ACL requiere especial atención. Por ejemplo, si intenta eliminar una línea concreta de una ACL numerada como se muestra aquí, se eliminará toda la ACL.

```
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 y péguela en un servidor TFTP o en un editor de texto como Bloc de notas para modificar las ACL numeradas. Después, realice los cambios deseados, copie la configuración y péguela en el router.

También puede hacer lo siguiente.

```
router#configure terminal
 Enter configuration commands, one per line.
 router(config)#ip access-list extended test
 router(config-ext-nacl)#permit ip host 2.2.2.2 host 3.3.3.3
 router(config-ext-nacl)#permit tcp host 1.1.1.1 host 5.5.5.5 eq www
 router(config-ext-nacl)#permit icmp any any
 router(config-ext-nacl)#permit udp host 6.6.6.6 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 2.2.2.2 host 3.3.3.3
    permit tcp host 1.1.1.1 host 5.5.5.5 eq www
    permit icmp any any
    permit udp host 6.6.6.6 10.10.10.0 0.0.0.255 eq domain
```

Todas las entradas eliminadas se borran de la ACL y todas las adiciones se agregan a su parte final.

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


              !--- Entrada de ACL eliminada

  router(config-ext-nacl)#no permit icmp any any


              !--- Entrada de ACL agregada

  router(config-ext-nacl)#permit gre host 4.4.4.4 host 8.8.8.8
  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 2.2.2.2 host 3.3.3.3
      permit tcp host 1.1.1.1 host 5.5.5.5 eq www
      permit udp host 6.6.6.6 10.10.10.0 0.0.0.255 eq domain
      permit gre host 4.4.4.4 host 8.8.8.8
```

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

Configure la ACL ampliada 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 la 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 de 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 172.162.2.9
   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 172.162.2.11
internetrouter(config-ext-nacl)#^Z

internetrouter#show access-lists
Extended IP access list 101
   10 permit tcp any any
   15 permit tcp any host 172.162.2.9
   18 permit tcp any host 172.162.2.11
   20 permit udp host 172.16.1.21 any
   30 permit udp host 172.16.1.22 any
internetrouter#
```

De igual modo, puede configurar la lista de acceso estándar de esta forma:

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

Este ejemplo muestra 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 201.101.110.0, wildcard bits 0.0.0.255
   25 deny any
```

Agregue la entrada a 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 a la parte superior de la lista para dar prioridad a la dirección IP específica en lugar de a 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 201.101.110.0, wildcard bits 0.0.0.255
   25 deny   any
```

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

Resolución de problemas

¿Cómo se elimina una ACL de una interfaz?

Acceda 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>
no ip access-group #in|out
```

¿Qué se puede hacer cuando se rechaza demasiado tráfico?

Si se rechaza demasiado tráfico, examine la lógica de la lista o intente definir y aplicar una lista más amplia. El comando **show ip access-lists** proporciona un recuento de paquetes en el que se indica qué entrada de ACL se utiliza.

La palabra clave **log**, al final de las entradas de ACL, indica el número de listas y si el paquete se ha permitido o rechazado, además de datos específicos del puerto.

Nota: la palabra clave **log-input** está presente en la versión 11.2 y posteriores del software Cisco IOS, así como en cierto software basado en la versión 11.1 de Cisco IOS creado específicamente para el mercado de proveedores de servicios. Este software anterior no admite esta palabra clave. Su uso incluye la interfaz de entrada y la dirección MAC de origen allí donde proceda.

¿Cómo se puede depurar en el nivel de los paquetes con un router de Cisco?

En este procedimiento se explica el proceso de depuración. Antes de empezar, compruebe que no se haya aplicado ninguna ACL en ese momento, que exista una ACL y que no esté inhabilitada la conmutación rápida.

Nota: tenga mucho cuidado al depurar un sistema con tráfico intenso. Use una ACL para depurar un tráfico específico. No obstante, debe estar seguro del proceso y del flujo de tráfico.

1. Ejecute el comando **access-list** para capturar los datos deseados.

   En este ejemplo, se ha configurado la captura de datos de la dirección de destino 10.2.6.6 o de la dirección de origen 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 la conmutación rápida en las interfaces involucradas. Sólo podrá ver el primer paquete si está habilitada la conmutación rápida.

```
                  config interface
no ip route-cache
```

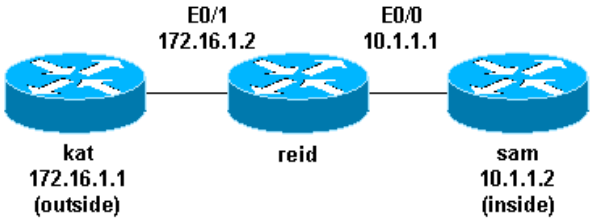
3. Ejecute el comando **terminal monitor** en modo habilitado para mostrar el resultado del comando **debug** y mensajes de error del sistema del terminal y la sesión actuales.
4. Ejecute el comando **debug ip packet 101** o el comando **debug ip packet 101 detail** para empezar el proceso de depuración.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

```
                  config interface
ip route-cache
```

Tipos de ACL IP

Esta sección del documento describe los tipos de ACL.

Diagrama de la red



ACL estándar

Las ACL estándar son el tipo más antiguo de ACL. Su aparición se remonta a la versión 8.3 del software Cisco IOS. Las ACL estándar controlan el tráfico por medio de la comparación de la dirección de origen de los paquetes IP con las direcciones configuradas en la ACL.

Este es el formato de la sintaxis de los comandos de una ACL estándar.

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

En todas las versiones del software, *access-list-number* puede ser cualquier número del 1 al 99. En la versión 12.0.1 del software Cisco IOS, las ACL estándar empiezan a usar más números (del 1300 al 1999). Estos números adicionales se denominan ACL IP ampliadas. En la versión 11.2 del software Cisco IOS se añadió la posibilidad de utilizar un valor *name* de lista en las ACL estándar.

Una configuración *source/source-wildcard* de 0.0.0.0/255.255.255.255 puede especificarse como **any**. El comodín puede omitirse si está formado sólo por ceros. Por consiguiente, el host 10.1.1.2 0.0.0.0 es igual al host 10.1.1.2.

Después de definir la ACL, se debe aplicar a la interfaz (entrante y saliente). En versiones anteriores del software, "out" era el valor predeterminado cuando la palabra clave "out" o "in" no se había especificado. En las versiones posteriores del software, se debe especificar la dirección.

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

Este es un ejemplo del uso de una ACL estándar para bloquear todo el tráfico excepto el procedente del origen 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 ampliadas

Las ACL ampliadas se introdujeron en la versión 8.3 del software Cisco IOS. Controlan el tráfico por medio de 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 la sintaxis de los comandos de las ACL ampliadas. Se han añadido saltos de línea por cuestiones 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-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 del software, *access-list-number* puede ser del 101 al 199. En la versión 12.0.1 del software Cisco IOS, las ACL ampliadas empiezan a usar más números (del 2000 al 2699). Estos números adicionales se denominan ACL IP ampliadas. En la versión 11.2 del software Cisco IOS se añadió la posibilidad de utilizar un valor *name* de lista en las ACL ampliadas.

El valor de 0.0.0.0/255.255.255.255 se puede especificar como **any**. Después de definir la ACL, se debe aplicar a la interfaz (entrante y saliente). En versiones anteriores del software, "out" era el valor predeterminado cuando la palabra clave "out" o "in" no se había especificado. En las versiones posteriores del software, se debe especificar la dirección.

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

Esta ACL ampliada sirve para permitir el tráfico en la red 10.1.1.x (interna) y para recibir respuestas de ping de fuera a la vez que impide los pings no solicitados de usuarios externos (aunque permite el resto del 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, por ejemplo, la administración de red, necesitan pings para una función de señal de mantenimiento. Si éste fuera el caso, sería mejor que limitara los pings de bloqueo entrantes o que fuera más detallado en relación con los IP permitidos/denegados.

Lock-and-Key (ACL dinámicas)

La función Lock-and-Key (también conocida como ACL dinámicas) apareció en la versión 11.1 del software Cisco IOS. Esta función depende de Telnet, de la autenticación (local o remota) y de las ACL ampliadas.

La configuración de Lock-and-Key comienza con la aplicación de una ACL ampliada para bloquear el tráfico que pasa por el router. La ACL ampliada bloquea a los usuarios que desean pasar por el router hasta que establecen una conexión desde Telnet al router y son autenticados. Luego, se pierde la conexión Telnet y se agrega una ACL dinámica de una única entrada a la ACL ampliada existente. De esta forma, se permite el tráfico durante un tiempo concreto; se admiten tiempos de espera absolutos e inactivos.

Este es el formato de la sintaxis de los comandos para la configuración de Lock-and-Key con autenticación local.

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

Después de la autenticación, se agrega dinámicamente la ACL de una única entrada a la ACL existente.

```
            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
```

A continuación se muestra un ejemplo básico de Lock-and-Key.

```
            username test password 0 test

               !--- Diez (minutos) es el tiempo de espera inactivo.

            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 (minutos) es el tiempo de espera absoluto.

            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
```

Cuando el usuario de 10.1.1.2 se conecta con Telnet a 10.1.1.1, se aplica 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 aparecieron en la versión 11.2 del software Cisco IOS. Así se permite que las ACL estándar y ampliadas reciban nombres en vez de números.

Este es el formato de la sintaxis de los comandos de las ACL con nombre IP.

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

A continuación se muestra 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]
```

En este ejemplo se ilustra el 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 la versión 11.3 del software Cisco IOS. Permiten filtrar los paquetes IP según los datos de sesión de capa superior. Suelen utilizarse para permitir el tráfico saliente y para limitar el tráfico entrante como respuesta a sesiones que se originan dentro del router.

Las ACL reflexivas sólo pueden ser definidas junto con las ACL con nombre IP ampliadas. No se pueden definir con ACL con nombre IP numeradas o estándar, ni con ACL de otros protocolos. Las ACL reflexivas pueden utilizarse conjuntamente con otras ACL ampliadas estándar y estáticas.

Esta es la sintaxis de los diversos comandos de las ACL reflexivas.

```
interface
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
```

En este ejemplo se ilustra cómo se permite el tráfico ICMP saliente y entrante, a la vez que sólo se permite el tráfico TCP iniciado desde dentro (el resto del tráfico se rechaza).

```
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


!--- Así se vincula la parte de la ACL reflexiva de la ACL de filtros de salida (outboundfilters),
!--- llamada "tcptraffic", a la ACL de filtros de entrada (inboundfilters).


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 la versión 12.0.1.T del software Cisco IOS. Aunque su función es similar a la de las ACL ampliadas, permiten el control de acceso según el tiempo. A fin de implementar ACL basadas en tiempo, se crea un intervalo de tiempo que define momentos específicos del día y la semana. El intervalo de tiempo se identifica con un nombre y se hace referencia a él a través de una función. Por lo tanto, las restricciones de tiempo vienen impuestas por la propia función. El intervalo temporal depende del reloj del sistema del router. El reloj del router se puede utilizar, pero la característica funciona mejor con la sincronización del Protocolo de tiempo de red (NTP).

Estos son comandos de las ACL basadas en tiempo.

```
   !--- Define un intervalo de tiempo con nombre.

time-range time-range-name

   !--- Define los momentos periódicos.

periodic days-of-the-week hh:mm to [days-of-the-week] hh:mm

   !--- O bien, define los tiempos absolutos.

absolute [start time date] [end time date]

   !--- El intervalo de tiempo utilizado en la ACL real.

ip access-list name|number <extended_definition>time-rangename_of_time-range
```

En este ejemplo, se permite una conexión Telnet de la red interna a la externa los lunes, miércoles y viernes en horario laborable:

```
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 presentaron en la versión 12.0.2.T del software Cisco IOS. Los comentarios facilitan la comprensión de las ACL y sirven para las ACL IP estándar o ampliadas.

Esta es la sintaxis de los comandos de las ACL IP con nombre comentadas.

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

Y ésta la de los comandos de las ACL IP numeradas comentadas.

```
access-list access-list-number remark remark
```

Aquí se ofrece un ejemplo de comentario 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 el contexto

El control de acceso basado en el contexto (CBAC) fue presentado en la versión 12.0.5.T del software Cisco IOS y requiere del conjunto de funciones de firewall de Cisco IOS. CBAC inspecciona el tráfico que discurre por el firewall para descubrir y administrar datos de estado de las sesiones TCP y UDP. Estos datos de estado sirven para crear aperturas temporales en las listas de acceso del firewall. Para ello, configure las listas de **ip inspect** en la dirección del flujo de inicio del tráfico para permitir el tráfico de retorno y conexiones de datos adicionales para sesiones aceptables (sesiones que se originaron dentro de la red interna protegida).

Esta es la sintaxis de CBAC.

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

En este ejemplo se ilustra el uso de CBAC para inspeccionar el tráfico saliente. La ACL ampliada 111 suele bloquear el tráfico de retorno (que no es ICMP) sin que CBAC abra agujeros para dicho tráfico.

```
          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 la versión 12.0.5.T del software Cisco IOS. Es imprescindible tener configurada la función de firewall de Cisco IOS. El proxy de autenticación sirve para autenticar usuarios de entrada, de salida o ambos. Los usuarios que suele bloquear una ACL pueden abrir un navegador Web para que atraviese el firewall y autenticarse en un servidor TACACS+ o RADIUS. El servidor envía entradas de ACL adicionales al router para permitir el paso a los usuarios después de la autenticación.

El proxy de autenticación es similar a la función Lock-and-Key (ACL dinámicas). Las diferencias entre ambos son las siguientes:

- La conexión Telnet al router activa la función Lock-and-Key. HTTP activa el proxy de autenticación a través del router.
- El proxy de autenticación tiene que usar un servidor externo.
- El proxy de autenticación puede gestionar la adición de varias listas dinámicas. Lock-and-Key sólo puede agregar una.
- El proxy de autenticación tiene un tiempo de espera absoluto, pero ninguno inactivo. Lock-and-Key tiene los dos tiempos de espera.

Consulte Cisco Secure Integrated Software Configuration Cookbook (Compendio de configuraciones de software de Cisco seguras e integradas) para ver ejemplos de proxy de autenticación.

ACL turbo

Las ACL turbo aparecieron en la versión 12.1.5.T del software Cisco IOS y sólo se encuentran en las plataformas 7200, 7500 y en otras de capacidad alta. La característica ACL turbo está diseñada para procesar las ACL más eficazmente con el fin de mejorar el rendimiento del router.

Use el comando **access-list compiled** para las ACL turbo. 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 estándar o ampliada, use el comando **global configuration** para compilar.

```
!--- Indica al router que compile

access-list compiled

Interface Ethernet0/1
ip address 172.16.1.2 255.255.255.0

!--- Se aplica a la interfaz

ip access-group 101 in
```

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

ACL basadas en tiempo distribuidas

Las ACL basadas en tiempo distribuidas aparecieron en la versión 12.2.2.T del software Cisco IOS para implementar las ACL basadas en tiempo en routers de la serie 7500 preparados para VPN. Antes de la aparición de la función de ACL basada en tiempo distribuida, las ACL basadas en tiempo no se admitían en las tarjetas de línea para los routers Cisco serie 7500. Si se configuraban, funcionaban como ACL normales. Si una interfaz en una tarjeta de línea se configuraba con ACL basadas en tiempo, los paquetes conmutados en la interfaz no se distribuían conmutados a través de la tarjeta de línea, pero se reenviaban al procesador de rutas para su procesamiento.

La sintaxis para las ACL distribuidas basadas en tiempo es la misma que la de las ACL basadas en tiempo, con la adición de comandos sobre el estado de los mensajes de comunicación entre procesadores (IPC) entre el procesador de rutas y la tarjeta de línea.

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

ACL de recepción

Las ACL de recepción sirven para aumentar la seguridad en los routers Cisco 12000 mediante la protección del procesador de rutas gigabit (GRP) del router frente al tráfico innecesario y potencialmente peligroso. Las ACL de recepción se añadieron como una renuncia especial al acelerador de mantenimiento de la versión 12.0.21S2 del software Cisco IOS y se integraron en la versión 12.0(22)S. Consulte GSR: Receive Access Control Lists (GSR: Listas de control de acceso de recepción) para obtener más información.

ACL de protección de infraestructuras

Las ACL de infraestructuras sirven para reducir al mínimo el riesgo y la eficacia de los ataques directos a la infraestructura, permitiendo explícitamente que acceda a ella sólo el tráfico autorizado y el resto de tráfico de tránsito. Consulte Protecting Your Core: Infrastructure Protection Access Control Lists (Protección del núcleo: Listas de control de acceso de protección de infraestructuras) para obtener más información.

ACL de tránsito

Las listas de control de acceso de tránsito sirven para aumentar la seguridad de la red puesto que permiten explícitamente sólo el tráfico requerido en ella. Consulte Transit Access Control Lists: Filtering at Your Edge (Listas de control de acceso de tránsito: Filtro por su lado) para obtener más información.

Fecha de Generación del PDF: 23 Junio 2008

http://www.cisco.com/cisco/web/support/LA/7/75/75923_confaccesslists.html