Configurar e filtrar listas de acesso IP

Contents

Introduction

Este documento descreve vários tipos de Listas de Controle de Acesso (ACLs - Access Control Lists) IP e como elas podem filtrar o tráfego de rede.

Prerequisites

Requirements

Não existem requisitos específicos para este documento. Os conceitos discutidos estão presentes no Cisco IOS® Software Releases 8.3 ou posterior. Isso é observado abaixo de cada recurso da lista de acesso.

Componentes Utilizados

Este documento discute vários tipos de ACL. Alguns estão presentes desde os Cisco IOS Software Releases 8.3. Já outros foram introduzidos em software releases posteriores. Isso é observado na discussão 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. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

Conventions

Consulte as Convenções de dicas técnicas da Cisco para obter mais informações sobre as convenções do documento.

Informações de Apoio

Este documento descreve como as listas de controle de acesso de IP podem filtrar o tráfego da rede. Também contém descrições breves dos tipos ACL IP, disponibilidade de recurso e um exemplo de uso em uma rede.

> **Note**: O RFC 1700 contém números atribuídos de portas conhecidas.O RFC 1918 contém alocação de endereço para Internet privadas, endereços IP que normalmente não devem ser vistos na Internet.

> **Note**: Somente usuários registrados da Cisco podem acessar informações internas.

> **Note**: As ACLs também podem ser usadas para definir o tráfego para o Network Address Translate (NAT), criptografar ou filtrar protocolos não IP, como AppleTalk ou IPX. Uma discussão sobre essas funções está fora do escopo deste documento.

Conceitos de ACL

Máscaras

As máscaras são usadas com endereços IP nas ACLs IP para especificar o que deve ser permitido e negado. As máscaras usadas para configurar endereços IP em interfaces começam com 255 e têm os grandes valores no lado esquerdo, por exemplo, endereço IP 10.165.202.129 com uma máscara 255.255.255.224. As máscaras para ACLs IP são o inverso, por exemplo, máscara 0.0.0.255. Às vezes, isso é chamado de máscara inversa ou máscara curinga. Quando o valor da máscara é dividido em binário (0s e 1s), os resultados determinam quais bits de endereço considerar quando o tráfego é processado. A 0 indica que os bits do endereço devem ser considerados (correspondência exata); um 1 na máscara é um *não importa*. Esta tabela explica melhor o conceito.

	Exemplo de máscara
endereço de rede (tráfego que deve ser processado)	*10.1.1.0*
máscara	*0.0.0.255*
endereço de rede (binário)	*00001010.00000001.00000001.00000000*
máscara (binária)	*00000000.00000000.00000000.11111111*

Com base na máscara binária, você pode ver que os primeiros três conjuntos (octetos) devem corresponder exatamente ao endereço binário de rede dado (00001010.00000001.00000001). O último conjunto de números é *não importa* (.11111111). Portanto, todo o tráfego que começa com 10.1.1. corresponde desde o último octeto é *não importa*. Portanto, com essa máscara, os endereços de rede de 10.1.1.1 a 10.1.1.255 (10.1.1.x) são processados.

Subtraia a máscara normal de 255.255.255.255 para determinar a máscara inversa da ACL. Neste exemplo, a máscara inversa é determinada para o endereço de rede 172.16.1.0 com uma máscara normal de 255.255.255.0.

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

Observe os equivalentes da ACL.

- A origem/caractere curinga de 0.0.0.0/255.255.255.255 significa **qualquer um.**
- A origem/caractere curinga de 10.1.1.2/0.0.0.0 é igual ao **host 10.1.1.2**.

Sumarização de ACLs

> **Note**: Máscaras de sub-rede também podem ser representadas por uma notação de comprimento fixo. Por exemplo, 192.168.10.0/24 representa 192.168.10.0 255.255.255.0.

Essa lista descreve como resumir um intervalo de redes em uma única rede para otimização da 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
```

Os dois primeiros e o último octeto são os mesmos para cada rede. Esta tabela é uma explicação

de como resumi-los em uma única rede.

O terceiro octeto para as redes anteriores pode ser escrito como visto nesta tabela, correspondente à posição do bit do octeto e ao valor do endereço 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

Como os primeiros cinco bits são correspondentes, as oito redes anteriores podem ser resumidas em uma rede (192.168.32.0/21 ou 192.168.32.0 255.255.248.0). Todas as oito combinações possíveis dos três bits de ordem baixa são relevantes para os intervalos de rede em questão. Esse comando define uma ACL que permite essa rede. Se você subtrair 255.255.248.0 (máscara normal) de 255.255.255.255, o resultado será 0.0.7.255.

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

Considere esse grupo de redes para uma explicação melhor.

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

Os dois primeiros e o último octeto são os mesmos para cada rede. Esta tabela é uma explicação de como resumi-los.

O terceiro octeto para as redes anteriores pode ser escrito como visto nesta tabela, correspondente à posição do bit do octeto e ao valor do endereço 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	?	?	?

Ao contrário do exemplo anterior, você não pode resumir essas redes em uma única rede. Se elas forem resumidas em uma única rede, serão transformadas em 192.168.144.0/21, pois há cinco bits semelhantes no terceiro octeto. Essa rede sumarizada 192.168.144.0/21 abrange um intervalo de redes de 192.168.144.0 a 192.168.151.0. Entre elas, 192.168.144.0, 192.168.145.0, 192.168.150.0 e 192.168.151.0 as redes não estão na lista fornecida de quatro redes. Para cobrir as redes específicas em questão, você precisa de no mínimo duas redes resumidas. Essas quatro redes podem ser resumidas nestas duas redes:

- Para as redes 192.168.146.x e 192.168.147.x, todos os bits correspondem, exceto o último, que é um *não importa*. Esse bit pode ser gravado como 192.168.146.0/23 (ou 192.168.146.0

255.255.254.0).
- Para as redes 192.168.148.x e 192.168.149.x, todos os bits correspondem, exceto o último, que é um *não importa*. Esse bit pode ser gravado como 192.168.148.0/23 (ou 192.168.148.0 255.255.254.0).

Essa saída define uma ACL sumarizada para as 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
```

ACLs de Processo

O tráfego que entra no roteador é comparado às entradas de ACL com base na ordem em que as entradas ocorrem no roteador. São adicionadas novas instruções no final da lista. O roteador continua procurando até encontrar uma correspondência. Se nenhuma correspondência for encontrada quando o roteador atingir o final da lista, o tráfego será negado. Por esse motivo, você deve ter as entradas frequentemente acessadas no início da lista. O tráfego que não é permitido é negado de forma implícita. Uma ACL de entrada única com apenas uma entrada deny pode negar todo o tráfego. É necessário haver pelo menos uma instrução de permissão em uma ACL. Caso contrário, todo o tráfego será bloqueado. Essas duas ACLs (101 e 102) têm o mesmo efeito.

```
!--- 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
```

No próximo exemplo, a última entrada é suficiente. Você não precisa das três primeiras entradas porque o IP inclui TCP, UDP e 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
```

Defina Portas e Tipos de Mensagem

Você não só pode definir a origem e o destino da ACL, como também pode definir portas, tipos de mensagens ICMP e outros parâmetros. Uma boa fonte de informações para portas bem conhecidas é o RFC 1700. Os tipos de mensagem ICMP são explicados no RFC 792 .

O roteador pode exibir um texto descritivo sobre algumas das portas conhecidas. Use um**?**para obter ajuda.

```
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 a configuração, o roteador também converte valores numéricos em valores mais amigáveis. Este é um exemplo onde você digita o número do tipo de mensagem ICMP e faz com que o roteador converta o número em um nome.

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

torna-se

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

ACLs de Aplicação

Você pode definir ACLs e ainda não aplicá-las. Entretanto, as ACLs não causarão nenhum efeito até que sejam aplicadas à interface do roteador. Convém aplicar a ACL na interface o mais próximo possível da origem do tráfego. Como mostrado neste exemplo, ao tentar bloquear o tráfego da origem para o destino, você pode aplicar uma ACL de entrada para E0 no roteador A em vez de uma lista de saída para E1 no roteador C. Uma lista de acesso tem **adeny ip any** any implicitamente no final de qualquer lista de acesso. Se o tráfego estiver relacionado a uma solicitação DHCP e não for explicitamente permitido, ele será descartado porque quando você examinar a solicitação DHCP no IP, o endereço origem será s=0.0.0.0 (Ethernet1/0), d=255.255.255.255, len 604, rcvd 2 UDP src=68, dst=67. Observe que o endereço IP origem é 0.0.0.0 e o endereço destino é 255.255.255 255. A porta de origem é 68 e o destino 67. Portanto, você deve permitir esse tipo de tráfego em sua lista de acesso ou o tráfego será descartado devido à negação implícita no final da instrução.

> **Note**: Para que o tráfego UDP passe, o tráfego UDP também deve ser permitido explicitamente pela ACL.

Defina Entrada, Saída, Origem e Destino

O roteador usa os termos de entrada, saída, origem e destino como referência. O tráfego no roteador pode ser comparado ao tráfego na via. Se você fosse um policial na Pensilvânia e quisesse parar um caminhão que viaja de Maryland a Nova York, a origem do caminhão é Maryland e o destino do caminhão é Nova York. O bloqueio poderia ser aplicado na fronteira Pensilvânia-Nova Iorque (out) ou na fronteira Maryland-Pensilvânia (in).

Esses são os significados de tais termos quando se trata de um roteador.

- **Saída: tráfego que já passou pelo roteador e sai da interface.** A origem é por onde o tráfego passou, no outro lado do roteador, e o destino é para onde ele vai.
- **Entrada: tráfego que chega na interface e, em seguida, passa pelo roteador.** A origem é por onde o tráfego passou e o destino é para onde ele vai, no outro lado do roteador.
- **Entrada: se a lista de acesso for de entrada, quando o roteador receber um pacote, o Cisco IOS Software verifica as instruções dos critérios da lista de acesso de uma correspondência.** Se o pacote for permitido, o software continuará a processá-lo. Se o pacote for negado, o software o descartará.
- **Saída: se a lista de acesso for de saída, depois que o software receber e rotear um pacote para a interface de saída, ele verificará as instruções dos critérios da lista de acesso de uma correspondência.** Se o pacote for permitido, o software o transmitirá. Se o pacote for negado, o software o descartará.

A ACL de entrada tem uma origem em um segmento da interface à qual é aplicada e um destino direcionado para quaisquer outras interfaces. A ACL de saída tem uma origem em um segmento de qualquer interface que não seja a interface à qual é aplicada e um destino que sai da interface à qual é aplicada.

ACLs de Edição

Ao editar uma ACL, é necessário ter atenção especial. Por exemplo, se você quiser excluir uma linha específica de uma ACL numerada existente, conforme mostrado aqui, a ACL inteira será excluída.

```
!--- 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 a configuração do roteador para um servidor TFTP ou um editor de texto, como o Bloco de Notas, para editar ACLs numeradas. Em seguida, faça todas as alterações e copie a configuração novamente para o roteador.

Você também pode fazer isso.

```
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 as exclusões são removidas da ACL e quaisquer adições são feitas no final da 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
```

Você também pode adicionar linhas de ACL às ACLs numeradas padrão ou numeradas estendidas pelo número de sequência no Cisco IOS. Veja um exemplo de configuração:

Configure a ACL estendida desta maneira:

```
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#
```

Execute o comando **show access-**list para visualizar as entradas da ACL. Os números de sequência, como 10, 20 e 30, também aparecem aqui.

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

Adicione a entrada para a lista de acesso 101 com o número de sequência 5.

Exemplo 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#
```

Na saída do comando **show access-**list, a ACL de número de sequência 5 é adicionada como a primeira entrada à lista de acesso 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#
```

Exemplo 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 semelhante, você pode configurar a lista do acesso padrão desta maneira:

```
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
```

A principal diferença em uma lista de acesso padrão é que o Cisco IOS adiciona uma entrada na ordem descendente do endereço IP, não em um número de sequência.

Este exemplo mostra as entradas diferentes, por exemplo, como permitir um endereço IP (192.168.100.0) ou as 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
```

Adicione a entrada da lista de acesso 2 para permitir o endereço IP 172.22.1.1:

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

Essa entrada é adicionada à parte superior da lista para dar prioridade ao endereço IP específico, e não à rede.

```
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
```

Note: As ACLs anteriores não são suportadas em mecanismos de segurança, tais como o firewall ASA/PIX.

Diretrizes para alterar listas de acesso quando elas são aplicadas a mapas de criptografia

- Se você adicionar a uma configuração de lista de acesso atual, não há necessidade de remover o mapa de criptografia. As adições diretamente a elas sem remover o mapa de criptografia são suportadas e aceitáveis.
- Se precisar modificar ou excluir uma entrada de lista de acesso de uma lista de acesso atual, você deverá remover o mapa de criptografia da interface. Depois de remover o mapa de criptografia, faça todas as alterações na lista de acesso e adicione o mapa de criptografia novamente. Alterações como a exclusão da lista de acesso sem remover o mapa de criptografia não são suportadas e podem resultar em comportamentos imprevisíveis.

Troubleshoot

Como eu removo um ACL de uma interface?

Vá para o modo de configuração e digite **no à frente do comando access-group, conforme mostrado neste exemplo, para remover uma ACL de uma interface.**

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

O que fazer quando muito tráfego é negado?

Se muito tráfego for negado, estude a lógica de sua lista ou tente definir e aplicar outra lista mais ampla. O **comando show ip access-lists fornece uma contagem de pacotes que mostra qual entrada de ACL é pressionada.** A palavra-chave do log na extremidade das entradas de ACL individuais mostra o número de ACL e também se o pacote foi permitido ou negado, além de informações específicas da porta.

Note: A palavra-chave de entrada do registro existe no software Cisco IOS versão 11.2 e posterior, e em determinados softwares com base no Cisco IOS Software Release 11.1 criados especificamente para o mercado de provedor de serviço. Software mais antigo não suporta essa palavra-chave. O uso desta palavra-chave inclui a interface de entrada e o endereço MAC de origem, quando aplicável.

Como debugar no nível do pacote que usa um Cisco Router?

Este procedimento explica o processo de debugação. Antes de começar, tenha certeza de que não haja ACLs aplicadas no momento, de que haja uma ACL e de que o fast switching não esteja desabilitado.

Note: Tenha muito cuidado ao debugar um sistema com tráfego intenso. Use uma ACL para debugar um tráfego específico. Mas garanta o processo e o fluxo de tráfego.

1. Use o comando **access-**list para capturar os dados desejados.Neste exemplo, a captação de

dados é definida para o endereço de destino de 10.2.6.6 ou o endereço de origem 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. Desative a switching rápida nas interfaces envolvidas. Você só verá o primeiro pacote se o fast switching não estiver desabilitado.

```
configure terminal
interface
```

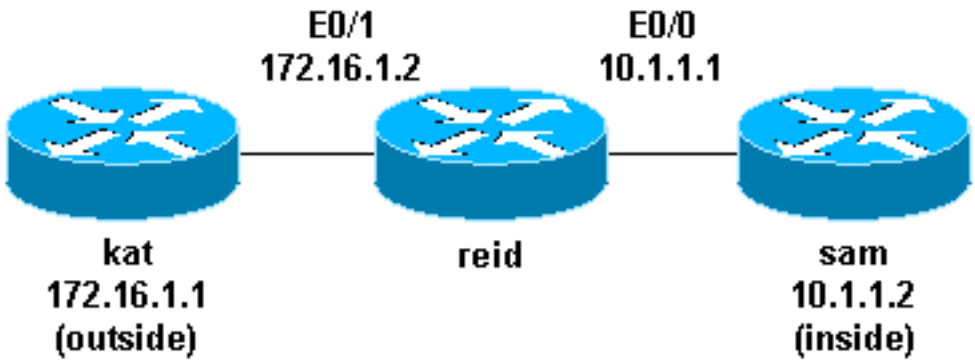
3. Use o **comando terminal monitor no modo enable para exibir a saída do comando debug e as mensagens de erro do sistema do terminal e da sessão atuais.**
4. Use o comando **debug ip packet 101** ou **debug ip packet 101 detail** para iniciar o processo de depuração.
5. Execute o **comando no debug all no modo enable e o comando interface configuration para interromper o processo de debugação.**
6. Reinicie o cache.

```
configure terminal
interface
```

Tipos de IP ACLs

Esta seção do documento descreve os tipos de ACL.

Diagrama de Rede



ACLs padrões

As ACLs padrão são o tipo mais antigo de ACL. Eles remontam ao Cisco IOS Software Release 8.3. As ACLs padrão controlam o tráfego pela comparação do endereço de origem dos pacotes IP com os endereços configurados na ACL.

Este é o formato da sintaxe do comando de uma ACL padrão.

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

Em todas as versões do software, o *número da lista de acesso* pode ser qualquer um de 1 a 99.

No Cisco IOS Software Release 12.0.1, as ACLs padrão começam a usar números adicionais (1300 a 1999). Esses números adicionais são mencionados como ACLs de IP expandidos. O Cisco IOS Software Release 11.2 acrescentou a capacidade de usar nomes de lista em ACLs padrão.

Uma configuração *s source/source-wildcard* de 0.0.0.0/255.255.255.255 pode ser especificada como **any** . O caractere curinga poderá ser omitido se for zero. Portanto, o host 10.1.1.2 0.0.0.0 é o mesmo que o host 10.1.1.2.

Depois que o ACL estiver definido, ele deverá ser aplicado à interface (de entrada ou saída). Em releases anteriores do software, saída era o padrão quando uma palavra-chave de saída ou de entrada não era especificada. A direção deve ser especificada nas versões posteriores do software.

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

Este é um exemplo do uso de uma ACL padrão para bloquear todo o tráfego, exceto o proveniente da origem 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
```

ACLs estendidos

As ACLs estendidas foram introduzidas no Cisco IOS Software Release 8.3. As ACLs estendidas controlam o tráfego pela comparação dos endereços de origem e destino dos pacotes IP com os endereços configurados na ACL.

Este é o formato da sintaxe do comando das ACL estendidas. As linhas são quebradas aqui para considerações de espaço.

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]
```

Em todas as versões de software, o *número da lista de acesso* pode ser 100 a 199. No Cisco IOS Software Release 12.0.1, as ACLs estendidas começam a usar números adicionais (2000 a 2699). Esses números adicionais são mencionados como ACLs de IP expandidos. O Cisco IOS Software versão 11.2 incluiu a capacidade de utilizar o nome da lista nas ACLs estendidas.

Épossível especificar o valor de 0.0.0.0/255.255.255.255 como **any.** Depois que o ACL estiver definido, ele deverá ser aplicado à interface (de entrada ou saída). Em releases anteriores do software, saída era o padrão quando uma palavra-chave de saída ou de entrada não era especificada. A direção deve ser especificada nas versões posteriores do software.

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

Essa ACL estendida é usada para permitir o tráfego na rede 10.1.1.x (interna) e para receber respostas de ping de fora, enquanto evita pings não solicitados de pessoas de fora, o que permite qualquer outro tráfego.

```
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
```

> **Note**: Alguns aplicativos, como gerenciamento de redes, exigem pings para uma função de keepalive. Se esse for o caso, você poderá limitar os pings de entrada bloqueados ou ser mais granular em IPs permitidos/negados.

Chave e bloqueio (ACLs dinâmicos)

Lock and key, também conhecido como ACLs dinâmicas, foi introduzido no Cisco IOS Software

Release 11.1. Esse recurso depende do Telnet, da autenticação (local ou remota) e das ACLs estendidas.

A configuração de bloqueio e chave tem início com a aplicação de uma ACL estendida para bloquear o tráfego no roteador. Os usuários que querem atravessar o roteador são bloqueados pela ACL estendida até que façam Telnet para o roteador e sejam autenticados. A conexão Telnet cai, e uma ACL dinâmica de entrada única é adicionada à ACL estendida existente. Isto permite o tráfego por um período específico; é possível ficar ocioso ou fazer intervalos absolutos.

Este é o formato da sintaxe do comando para a configuração de bloqueio e chave com autenticação local.

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

A ACL de entrada única nesse comando é adicionada de forma dinâmica à ACL existente após a autenticação.

```
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 é um exemplo básico de bloqueio e chave.

```
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
```

Depois que o usuário em 10.1.1.2 fizer uma conexão Telnet para 10.1.1.1, a ACL dinâmica será aplicada. Em seguida, a conexão é descartada e o usuário pode seguir para a rede 172.16.1.x.

ACLs de IP nomeados

As ACLs IP com nome foram introduzidas no Cisco IOS Software Release 11.2. Isso permite que as ACLs padrão e estendidas recebam nomes em vez de números.

Este é o formato da sintaxe do comando para ACLs de IP nomeadas.

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

Veja este exemplo 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 é um exemplo do uso de uma ACL nomeada para bloquear todo o tráfego, exceto a conexão Telnet do host 10.1.1.2 para o 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
```

ACLs reflexivos

As ACLs reflexivas foram introduzidas no Cisco IOS Software Release 11.3. As ACLs reflexivas permitem que os pacotes IP sejam filtrados com base nas informações de sessão da camada superior. Elas são usadas geralmente para permitir o tráfego de saída e para limitar o tráfego de entrada em resposta às sessões originadas dentro do roteador.

ACLs reflexivas podem ser definidas apenas com ACLs IP nomeadas estendidas. Elas não podem ser definidas com ACLs numeradas ou de IP nomeadas padrão nem com outras ACLs de protocolo. ACLs reflexivas podem ser usadas em conjunto com outras ACLs padrão e estáticas estendidas.

Esta é a sintaxe de vários comandos de ACL reflexiva.

```
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 é um exemplo da permissão do tráfego de saída e de entrada ICMP, enquanto ele permite apenas o tráfego TCP iniciado internamente, outros tráfegos são negados.

```
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
```

ACLs com base no tempo usando intervalos de tempo

ACLs com base em tempo foram introduzidas na versão do Cisco IOS Software Release 12.0.1.T. Embora sejam semelhantes aos ACLs estendidos com relação à função, eles permitem controle de acesso com base no tempo. É criado um intervalo de tempo que define horários específicos do dia e da semana para implementar ACLs com base em tempo. O intervalo de tempo é identificado por um nome e, em seguida, referenciado por uma função. Portanto, as restrições de tempo são impostas na própria função. O intervalo de tempo se baseia no relógio do sistema do roteador. O relógio do roteador pode ser utilizado, mas o recurso funciona melhor com a sincronização do NTP (Protocolo de tempo de rede).

Estes são comandos de ACL com base em tempo.

```
!--- 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
```

Neste exemplo, uma conexão Telnet é permitida da rede interna à rede externa às segundas-feiras, às quartas-feiras e às sextas-feiras durante o horário 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 IP ACL comentadas

Entradas de ACL IP comentadas foram introduzidas no Cisco IOS Software Release 12.0.2.T. Os comentários facilitam o entendimento das ACLs e podem ser usados para ACLs de IP padrão ou estendidas.

Esta é a sintaxe comentada do comando da ACL de IP nomeada.

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

Esta é a sintaxe comentada do comando da ACL de IP numerada.

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

Este é um exemplo de comentários dentro de uma 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
```

Controle de acesso baseado em contexto

O CBAC (Controle de acesso baseado em contexto) foi introduzido no Cisco IOS Software Release 12.0.5.T e requer o conjunto de recursos do Cisco IOS Firewall. O CBAC inspeciona o tráfego que passa pelo firewall para descobrir e gerenciar informações de estado para sessões de TCP e de UDP. Essas informações de estado são usadas para criar aberturas temporárias nas listas de acesso do firewall. Configure o ip inspetlists na direção do fluxo de iniciação de tráfego para permitir o tráfego de retorno e conexões de dados adicionais para sessões permissíveis, sessões originadas na rede interna protegida, para fazer isso.

Esta é a sintaxe do CBAC.

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

Este é um exemplo do uso do CBAC para inspecionar o tráfego de saída. A ACL estendida 111 normalmente bloqueia o tráfego de retorno que não seja de ICMP sem que o CBAC abra brechas para o tráfego 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 autenticação

O proxy de autenticação foi introduzido no Cisco IOS Software versão 12.0.5.T. Para isso, é necessário que o recurso Cisco IOS Firewall esteja configurado. O proxy de autenticação é usado para autenticar usuários de entrada ou saída ou ambos. Os usuários que normalmente são bloqueados por uma ACL podem iniciar uma sessão do navegador para passar pelo firewall e fazer a autenticação em um servidor RADIUS ou TACACS+. O servidor transmite entradas de

ACL adicionais ao roteador para permitir a passagem dos usuários após a autenticação.

O proxy de autenticação é semelhante à chave e bloqueio (ACLs dinâmicos). Estas são as diferenças:

- O bloqueio e chave é ativado por uma conexão Telnet para o roteador. O proxy de autenticação é ativado pelo HTTP através do roteador.
- O proxy de autenticação deve usar um servidor externo.
- O proxy de autenticação pode gerenciar a adição de várias lista dinâmicas. O bloqueio e chave pode adicionar somente uma lista dinâmica.
- O proxy de autenticação tem um timeout absoluto, mas não tem um timeout ocioso. O bloqueio e chave tem ambos.

Consulte o Cisco Secure Integrated Software Configuration Cookbook (Cookbook de configuração do software Cisco Secure Integrated) para obter exemplos de proxy de autenticação.

Turbo ACLs

As ACLs Turbo foram introduzidas no Cisco IOS Software Release 12.1.5.T e são encontradas somente no 7200, 7500 e em outras plataformas de ponta. O recurso ACL turbo foi projetado para processar ACLs de maneira mais eficiente para aprimorar o desempenho do roteador.

Use o **comando access-list compiled para ACLs turbo.** Este é um exemplo de uma 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
```

Depois que a ACL padrão ou estendida é definida, use o **comando global configuration para compilar.**

```
!--- 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
```

O **comando show access-list compiled exibe estatísticas referentes à ACL.**

ACLs com base em tempo distribuídas

As ACLs distribuídas com base em tempo foram introduzidas no Cisco IOS Software Release 12.2.2.T para implementar ACLs com base em tempo em roteadores 7500 Series habilitados por VPN. Antes da introdução do recurso de ACL distribuída com base em tempo, as ACL com base em tempo não eram suportadas em placas de linha para os roteadores Cisco 7500 Series. Se os ACLs foram configurados com base no tempo, eles se comportarão como ACLs normais. Se uma interface em uma placa de linha foi configurada com ACLs com base em tempo, os pacotes comutados na interface não foram comutados de forma distribuída pela placa de linha, e sim encaminhados para o processador de rota para processamento.

A sintaxe para ACLs distribuídas com base no tempo é a mesma usada para ACLs com base no tempo, com a adição dos comandos relacionados ao status das mensagens de comunicação entre processadores (IPC) entre o processador de rota e a placa de linha.

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

ACLs de Recebimento

ACLs de recebimento são usadas para aumentar a segurança em roteadores Cisco 12000 pela proteção do GRP do roteador do tráfego desnecessário e possivelmente prejudicial. ACLs de recebimento foram adicionadas como uma renúncia especial à obstrução de manutenção para o Cisco IOS Software Release 12.0.21S2 e integradas ao 12.0(22)S. Consulte o GSR: Receba listas de controle de acesso para obter mais informações.

ACLs de Proteção de Infraestrutura

As ACLs de infraestrutura são usadas para minimizar o risco e a eficácia do ataque direto à infraestrutura com a permissão explícita apenas do tráfego autorizado para o equipamento de infraestrutura, enquanto permite todo o tráfego restante. Consulte Proteção de Sua Base: Listas de Controle de Acesso de Proteção da Infraestrutura para obter mais informações.

ACLs de Trânsito

ACLs de trânsito são usadas para aumentar a segurança de rede, pois permitem de forma explícita somente o tráfego necessário em sua(s) rede(s). Consulte Listas de Controle de Acesso de Trânsito: Filtragem em Sua Extremidade para obter mais informações.

Informações Relacionadas

- Configurar ACLs de IP usadas com frequência
- RFC 1700
- RFC 1918
- Página de Suporte das Listas de Acesso
- Cisco IOS Firewall
- Suporte Técnico e Documentação - Cisco Systems

Sobre esta tradução
A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês (link fornecido) seja sempre consultado.