配置 IP 访问列表

目录

简介

本文档介绍了 IP 访问控制列表 (ACL) 如何能过滤网络流量。其中还包含 IP ACL 类型的简要描述、功能可用性以及网络使用示例。

访问软件顾问(仅限注册用户)工具为了确定某些的支持更加先进的Cisco IOS IP ACL功能。

RFC 1700 包含通用端口的指定编号。 RFC 1918 包含专用 Internet 的地址分配，通常是在 Internet 上不可见的 IP 地址。

注意： 除过滤 IP 数据流外，ACL 还可用于其他用途，例如，定义网络地址转换 (NAT) 或加密的数据流、或者过滤非 IP 协议（如 AppleTalk 或 IPX）。有关这些功能的详细信息不在本文档的讨论范围之内。

先决条件

要求

本文档没有任何特定的前提条件。讨论的概念是存在Cisco IOS软件版本8.3或以上。这在每个访问列表功能下都有标注。

使用的组件

本文档将讨论各种类型的 ACL。其中某些类型存在于 Cisco IOS 软件版本 8.3 及更高版本中，而其他类型则在更高的软件版本中引入。这一点在讨论每种类型时都有标注。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您使用的是真实网络，请确保您已经了解所有命令的潜在影响。

规则

有关文档规则的详细信息，请参阅 Cisco 技术提示规则。

ACL 概念

此部分阐述了 ACL 概念。

掩码

掩码与 IP 地址一起用于 IP ACL 中，以指定应该允许和拒绝哪些内容。用于配置接口 IP 地址的掩码以 255 开头，大的数值列于左边，例如，IP 地址 209.165.202.129 对应掩码为 255.255.255.224。IP ACL 的掩码则相反，如掩码 0.0.0.255。它有时称为反掩码或通配符掩码。如果将掩码的值分解为二进制（0s 和 1s），其结果可以确定处理数据流时所要考虑的地址位。A 0 表示这些地址位必须要考虑（完全匹配）；掩码中的 a 1 表示“忽略”。下表进一步说明了这一概念。

掩码示例	
网络地址（要处理的数据流）	*10.1.1.0*
掩码	*0.0.0.255*
网络地址（二进制）	*00001010.00000001.00000001.00000000*
掩码（二进制）	*00000000.00000000.00000000.11111111*

从二进制掩码可以看出，前三组（八位字节）必须与给定的二进制网络地址

(00001010.00000001.00000001) 完全匹配。最后一组数字为“忽略”(.11111111)。因此，所有以 10.1.1. 开头的数据流都视为匹配，因为最后一个八位组为“忽略”。因此，通过该掩码，从 10.1.1.1 到 10.1.1.255 (10.1.1.x) 的网络地址均可处理。

将 255.255.255.255 减去正常掩码可确定 ACL 反掩码。在本示例中，确定的反掩码将用于正常掩码为 255.255.255.0 的网络地址 172.16.1.0。

- 255.255.255.255 - 255.255.255.0（正常掩码）= 0.0.0.255（反掩码）

请注意以下这些 ACL 等量。

- 源地址/源地址通配符 0.0.0.0/255.255.255.255 表示“任意”。
- 源地址/通配符 10.1.1.2/0.0.0.0 与“主机 10.1.1.2”相同。

ACL 汇总

注意： 子网掩码也可表示为固定长度掩码。例如，192.168.10.0/24 代表 192.168.10.0 255.255.255.0。

此列表说明了如何将一系列网络汇总为单个网络以进行 ACL 优化。请考虑以下这些网络。

```
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
```

每个网络的前两个八位组和最后一个八位组都是相同的。该表说明了如何将这些网络汇总为单个网络。

按照八位组的位位置和每个位的地址值，可将以上网络的第三个八位组写成如下表所示。

十进制	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

由于前五位匹配，因此可将以上八个网络汇总为一个网络（192.168.32.0/21 或 192.168.32.0 255.255.248.0）。三个低位的所有八个可能的组合则与相关的网络范围有关。以下命令定义了一个允许该网络的 ACL。将 255.255.255.255 减去 255.255.248.0（正常掩码），即可得到 0.0.7.255。

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

请考虑以下这组网络以进行进一步说明。

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

每个网络的前两个八位组和最后一个八位组都是相同的。该表格说明了如何对这些进行汇总。

按照八位组的位位置和每个位的地址值，可将以上网络的第三个八位组写成如下表所示。

十进制	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	??	??	??

不同于前一个示例，您不能汇总这些网络到单个网络。如果他们汇总对单个网络，他们变为192.168.144.0/21，因为有五个位类似在第三个八位位组。这汇总的网络192.168.144.0/21报道范围从192.168.144.0的网络到192.168.151.0。在这些中， 192.168.144.0、192.168.145.0、192.168.150.0和192.168.151.0网络不在四网络给的列表。为了报道有问题的特定的网络，您需要至少两汇总的网络。给的四网络可以汇总到这两网络：

- 对于网络192.168.146.x和192.168.147.x，所有位配比除了最后一个，是“不关心”。可以将此写为 192.168.146.0/23（或 192.168.146.0 255.255.254.0）。
- 对于网络192.168.148.x和192.168.149.x，所有位配比除了最后一个，是“不关心”。可将此写为192.168.148.0/23（或 192.168.148.0 255.255.254.0）。

该输出可为上述网络定义一个汇总 ACL。

```
!--- 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
```

处理 ACL

进入路由器的数据流将按照 ACL 条目在路由器中的顺序与 ACL 条目相比较。新的语句会添加到列表末尾。路由器会继续查看，直到出现匹配。如果路由器到达列表末端时还未找到匹配，数据流将被拒绝。因此，应将经常发生匹配的条目置于列表顶部。对于不允许的数据流，将默认拒绝。如果单条目的 ACL 中仅包含一个拒绝条目，其作用将是拒绝所有数据流。您必须在 ACL 中至少包含一条允许语句，否则所有数据流都会受阻。这两个 ACL（101 和 102）的作用相同。

```
!--- 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
```

在本示例中，最后一个条目已足够。不需要前三个条目，因为 TCP 包括 Telnet，而 IP 包括 TCP、用户数据报协议 (UDP) 和 Internet 控制消息协议 (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
```

定义端口和消息类型

除定义 ACL 源地址和目标地址外，还可定义端口、ICMP 消息类型和其他参数。要了解有关通用端口的信息，请参阅 RFC 1700。 有关 ICMP 消息类型的说明，请参阅 RFC 792。

路由器可显示有关某些通用端口的描述性文本。请使用 **?**，以获得帮助。

```
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)
```

在配置过程中，路由器还会将数值转换为更加方便的值。以下示例中，键入 ICMP 消息类型编号后，路由器将此编号转换成了一个名称。

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

变为

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

应用 ACL

您可以定义 ACL 而无需将其应用。但在应用于路由器接口之前 ACL 并不会生效。较好的做法是，在与源数据流最近的接口上应用 ACL。如本例所示，当您尝试阻止数据流从源位置传输到目标位置时，可在路由器 A 的 E0 上应用入站 ACL，而不是在路由器 C 的 E1 上应用出站列表。默认情况下，每个访问列表在其末尾都会有 **deny ip any any**。如果数据流与 DHCP 请求有关，且未得到明确允许，则该数据流将会丢失，因为当您查看 IP 中的 DHCP 请求时，源地址为 s=0.0.0.0(Ethernet1/0)，d=255.255.255.255，len 604，rcvd 2 UDP src=68，dst=67。请注意，源 IP 地址为 0.0.0.0，而目标地址为 255.255.255.255。源端口为 68，目标端口为 67。因此，应该在访问列表中允许这种数据流，否则数据流会由于语句末尾的默认拒绝而丢失。

注意： 要使 UDP 数据流通过，ACL 还必须明确允许 UDP 数据流。

定义输入、输出、入站、出站、源和目标

路由器使用“输入”、“输出”、“源”和“目标”作为引用术语。路由器上的数据流可以比作高速公路上的交通。如果您是宾夕法尼亚州的一名执法人员，想拦住从马里兰州开往纽约的卡车，则卡车的源位置是马里兰，目标位置是纽约。路障可设在宾夕法尼亚州 – 纽约边界（出界）或马里兰州 – 宾夕法尼亚州边界（入界）。

涉及到路由器时，这些术语有以下几种含义。

- **输出 —** 已通过路由器且离开接口的数据流。源位置是它在路由器另一端曾经所在的位置，而目

标位置是它要到达的位置。

- **输入** — 到达接口并通过路由器的数据流。源位置是它曾经所在的位置，目标位置是在路由器另一端它要到达的位置。
- **入站** — 如果是入站访问列表，则当路由器接收到数据包时，Cisco IOS 软件将检查访问列表中的条件语句，看是否有匹配。如果数据包被允许，则软件将继续处理该数据包。如果数据包被拒绝，则软件会丢弃该数据包。
- **出站** — 如果是出站访问列表，则当软件到接收数据包并将群其路由至出站接口后，软件将检查访问列表中的条件语句，看是否有匹配。如果数据包被允许，则软件会发送该数据包。如果数据包被拒绝，则软件会丢弃该数据包。

输入 ACL 的源位置在其应用的接口分段上，目标位置在其他任意接口上。输出 ACL 的源位置在除其应用的接口之外的任意接口的分段上，目标位置在其应用的接口上。

编辑 ACL

编辑 ACL 时需要特别注意。例如，如果您打算从此处所示的编号 ACL 中删除特定的某一行，则整个 ACL 将被删除。

```
!--- 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
```

将路由器的配置复制到 TFTP 服务器或文本编辑器（如记事本），以编辑编号 ACL。然后进行任意更改并将配置复制回路由器。

您也可以这样操作。

```
router#configure terminal Enter configuration commands, one per line. router(config)#ip access-
list extended test !--- Permits IP traffic from 2.2.2.2 host machine to 3.3.3.3 host machine.
router(config-ext-nacl)#permit ip host 2.2.2.2 host 3.3.3.3 !--- Permits www traffic from
1.1.1.1 host machine to 5.5.5.5 host machine. router(config-ext-nacl)#permit tcp host 1.1.1.1
host 5.5.5.5 eq www !--- Permits icmp traffic from any to any network. router(config-ext-
nacl)#permit icmp any any !--- Permits dns traffic from 6.6.6.6 host machine to 10.10.10.0
network. 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
```

删除的任意内容将从 ACL 中移除，添加的任意内容将加至 ACL 末尾。

```
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 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
```

还可以按照 Cisco IOS 中的序列号将 ACL 行添加到标准编号 ACL 或扩展编号 ACL 中。以下是一个配置示例：

用以下方式配置扩展 ACL：

```
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#
```

发出 **show access-list** 命令以查看 ACL 条目。这里还显示了 10、20 和 30 等序列号。

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

为访问列表 101 添加序列号为 5 的条目。

示例 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#
```

在 **show access-list** 命令输出中，序列号为 5 的 ACL 被添加为访问列表 101 的第一个条目。

```
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#
```

示例 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#
```

同样，您还可以用下列方式配置标准访问列表：

```
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
```

标准访问列表的主要区别在于 Cisco IOS 是按 IP 地址的降序添加条目，而不是按序列号。

本示例显示了不同的条目，例如，如何允许 IP 地址 (192.168.100.0) 或网络 (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
```

在访问列表 2 中添加条目，以允许 IP 地址 172.22.1.1：

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

此条目添加到了列表顶部，目的是为特定 IP 地址（而不是网络）赋予优先级。

```
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
```

注意： 如 ASA/PIX 防火墙之类的安全设备不支持以上 ACL。

访问列表应用于加密映射时更改访问列表的指南

- 如果向现有的访问列表配置中增加条目，则无需删除加密映射。如果直接向其中增加条目且不删除加密映射，这也是受支持且可以接受的。

- 如果需要在现有的访问列表中修改或删除访问列表条目，则必须从接口删除加密映射。删除加密映射后，可对访问列表做任意更改并重新添加加密映射。如果做了更改（如删除访问列表）但未删除加密映射，则不受支持且会导致无法预测的行为。

故障排除

如何从接口删除 ACL？

要从接口删除 ACL，请进入配置模式并在 access-group 命令前输入 **no**，如本示例所示。

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

如果被拒绝的数据流过多该如何处理？

如果被拒绝的数据流过多，请检查列表的逻辑，或尝试定义并应用其他更宽松的列表。**show ip access-lists** 命令可提供数据包计数，其中显示匹配的 ACL 条目。

除特定于端口的信息外，各 ACL 条目末尾的 **log** 关键字还可显示 ACL 编号以及数据包是被允许还是被拒绝。

注意： 在 Cisco IOS 软件版本 11.2 及更高版本，以及专门为服务提供商市场开发的某些基于 Cisco IOS 软件版本 11.1 的软件中，则使用了 **log-input** 关键字。旧版本的软件不支持该关键字。此关键字的用法包括适用的输入接口和源 MAC 地址。

如何在使用 Cisco 路由器的数据包级别上进行调试？

以下程序说明了调试过程。在开始之前，请确保当前未应用 ACL，已有一个 ACL，而且未禁用快速交换。

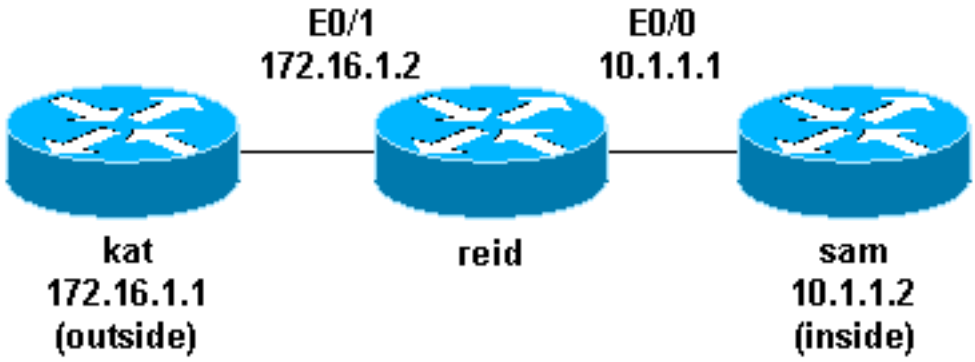
注意： 在调试流量较大的系统时请务必谨慎。请使用 ACL 调试特定的数据流。但是，请确定进程和数据流。

1. 请使用 **access-list** 命令捕获所需的数据。在本示例中，为目标地址 10.2.6.6 或源地址 10.2.6.6 设置了数据捕获。
   ```
   access-list 101 permit ip any host 10.2.6.6 access-list 101 permit ip host 10.2.6.6 any
   ```
2. 禁用相关接口上的快速交换。如果未禁用快速交换，您将只能看到第一个数据包。
   ```
   config interface no ip route-cache
   ```
3. 请在启用模式下使用 **terminal monitor** 命令，以显示 debug 命令的输出以及当前终端和会话的系统错误消息。
4. 请使用 **debug ip packet 101** 或 debug ip packet 101 detail 命令以开始调试进程。
5. 请在启用模式下执行 **no debug all** 命令和 interface configuration 命令以停止调试进程。
6. 重新启动高速缓存。
   ```
   config interface ip route-cache
   ```

IP ACL 的类型

本部分文档将介绍各种 ACL 类型。

网络图



标准 ACL

标准 ACL 是最早的 ACL 类型。它们可追溯到 Cisco IOS 软件版本 8.3。标准 ACL 通过将 IP 数据包源地址与 ACL 中配置的地址进行比较来控制数据流。

以下是标准 ACL 的命令语法格式。

access-list *access-list-number* **{permit|deny}** **{***host|source source-wildcard***|any}**
在所有软件版本中，*access-list-number* 可以是 1 到 99 的任何数。在 Cisco IOS 软件版本 12.0.1 中，标准 ACL 开始使用附加的编号（1300 到 1999）。这些附加的编号被称为扩展 IP ACL。Cisco IOS 软件版本 11.2 在标准 ACL 中增加了使用列表名称的功能。

source/source-wildcard 设置 0.0.0.0/255.255.255.255 可指定为 **any**。如果设置全部为零，则可以省略。因此，主机 10.1.1.2 0.0.0.0 与主机 10.1.1.2 相同。

定义 ACL 后，必须将其应用于接口（入站或出站）。在早期软件版本中，如果未指定关键字“输出”或“输入”，则默认为“输出”。在更高的软件版本中则必须指定该方向。

interface <*interface***> ip access-group number {in|out}**
以下是标准 ACL 的一个用例，目的是阻止来自源地址 10.1.1.x 以外的所有数据流。

```
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
```

扩展 ACL

扩展 ACL 是在 Cisco IOS 软件版本 8.3 中引入的。扩展 ACL 通过将 IP 数据包的源地址和目标地址与 ACL 中配置的地址进行比较来控制数据流。

以下是扩展 ACL 的命令语法格式。考虑到空间问题，这里作换行处理。

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***]**

在所有软件版本中， *access-list-number*可以是100到199。在 Cisco IOS 软件版本 12.0.1 中，扩展 ACL 开始使用附加的编号（2000 到 2699）。这些附加的编号被称为扩展 IP ACL。Cisco IOS 软件版本 11.2 在扩展 ACL 中增加了使用列表名称的功能。

值 0.0.0.0/255.255.255.255 可指定为 **any**。定义 ACL 后，必须将其应用于接口（入站或出站）。在早期软件版本中，如果未指定关键字“输出”或“输入”，则默认为“输出”。在更高的软件版本中则必须指定该方向。

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

该扩展 ACL 用于允许 10.1.1.x 网络（内部）上的数据流并接收外部 ping 响应，同时它还会阻止来自外部人员未经请求发送的 ping，而允许所有其他的数据流。

```
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
```

注意： 某些应用程序（如网络管理）需要使用 ping 以实现 keepalive 功能。这种情况下，您可能希望限制对入站 ping 的阻止或更详细地指定允许/拒绝的 IP。

锁和密钥（动态 ACL）

锁和密钥（亦称动态 ACL）是在 Cisco IOS 软件版本 11.1 中引入的。此功能依赖于 Telnet、身份验证（本地或远程）以及扩展 ACL。

锁和密钥配置从应用扩展 ACL 开始，以阻止数据流通过路由器。想要穿越路由器的用户将被扩展 ACL 阻止，直至 Telnet 到路由器并经过身份验证。然后 Telnet 连接会断开，一个单条目的动态 ACL 将会添加至已存在的扩展 ACL 中。这将在特定时间段允许数据流；可以设置空闲超时和绝对超时。

以下是含本地身份验证的锁和密钥配置的命令语法格式。

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

此命令中的单条目 ACL 将会在身份验证之后动态添加至已存在的 ACL 中。

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**
以下是锁和密钥的一个基本示例。

```
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
```

位于 10.1.1.2 的用户通过 Telnet 连接登录到 10.1.1.1 之后，动态 ACL 将得到应用。然后连接断开，用户即可进入 172.16.1.x 网络。

命名 IP ACL

命名 IP ACL 是在 Cisco IOS 软件版本 11.2 中引入的。它允许为标准和扩展 ACL 指定名称而不是编号。

以下是命名 IP ACL 的命令语法格式。

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

以下是 TCP 示例：

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

以下是命名 ACL 的一个用例，目的是阻止从主机 10.1.1.2 到主机 172.16.1.1 的 Telnet 连接以外的所有数据流。

```
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
```

自反 ACL

自反 ACL 是在 Cisco IOS 软件版本 11.3 中引入的。自反 ACL 允许根据上层会话信息过滤 IP 数据包。它们通常用于允许出站数据流，并限制对路由器内部发起的会话响应的入站数据流。

自反 ACL 只能通过扩展命名 IP ACL 进行定义，无法通过编号或标准命名 IP ACL 或其他协议 ACL 进行定义。自反 ACL 可与其他标准和静态扩展 ACL 结合使用。

以下是各种自反 ACL 命令的语法。

```
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
```

以下示例将允许 ICMP 出站和入站数据流，但只允许源自内部的 TCP 数据流，其他数据流都将被拒绝。

```
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
```

使用时间范围基于时间的 ACL

基于时间的 ACL 是在 Cisco IOS 软件版本 12.0.1.T 中引入的。尽管与扩展 ACL 在功能上类似，但它们可根据时间进行访问控制。为了实施基于时间的 ACL，需要创建指定每天和每周特定时间的时间范围。时间范围通过名称来定义，然后被相应的功能引用。因此，函数本身将会受到时间限制。时间范围依赖于路由器的系统时钟。可以使用路由器时钟，但是此功能与网络时间协议 (NTP) 同步搭配使用时效果最佳。

以下是基于时间的 ACL 命令。

```
!--- 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 <extended_definition>time-rangename_of_time-range
```

在本示例中，允许在星期一、星期三和星期五的营业时间内从内部网络到外部网络的 Telnet 连接：

```
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
```

带有注释的 IP ACL 条目

带有注释的 IP ACL 条目是在 Cisco IOS 软件版本 12.0.2.T. 中引入的。注释使 ACL 更加容易理解并且可用于标准或扩展 IP ACL。

以下是带有注释的命名 IP ACL 命令语法。

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

以下是带有注释的编号 IP ACL 命令语法。

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

以下示例将对编号 ACL 添加注释。

```
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
```

基于上下文的访问控制

基于上下文的访问控制 (CBAC) 是在 Cisco IOS 软件版本 12.0.5.T 中引入的，它需要使用 Cisco IOS 防火墙特性集。CBAC 会检查通过防火墙的数据流，找到并管理 TCP 和 UDP 会话的状态信息。使用此状态信息的目的是在防火墙访问列表中创建临时空缺。为此，请按数据流起始流向配置 **ip inspect** 列表，以允许返回的数据流和来自可允许会话（受保护内部网络发起的会话）的其他数据连接。

以下是 CBAC 的语法。

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

以下是使用 CBAC 检查出站数据流的示例。如果没有 CBAC 空缺位置给返回数据流，则扩展 ACL 111 通常会阻止除 ICMP 外的返回数据流。

```
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
```

身份验证代理

身份验证代理是在 Cisco IOS 软件版本 12.0.5.T 中引入的。它要求拥有 Cisco IOS 防火墙特性集。身份验证代理用于对入站或/和出站用户进行身份验证。通常被 ACL 阻止的用户可让浏览器通过防火墙并在 TACACS+ 或 RADIUS 服务器上进行身份验证。服务器会将附加 ACL 条目传递到路由器中，以便在身份验证后允许用户通过。

身份验证代理类似于锁和密钥（动态 ACL）。两者的区别如下：

- 锁和密钥通过到路由器的 Telnet 连接启用。身份验证代理则是通过 HTTP 在路由器上启用。
- 身份验证代理必须使用外部服务器。
- 通过身份验证代理可添加多个动态列表。锁和密钥只能添加一个。
- 身份验证代理有绝对超时，但没有空闲超时。锁和密钥两者均有。

请参阅 Cisco 安全集成软件配置手册，查看身份验证代理的示例。

Turbo ACL

Turbo ACL 是在 Cisco IOS 软件版本 12.1.5.T 中引入的，应用于 7200、7500 和其他高端平台上。Turbo ACL 功能旨在更有效地处理 ACL 以提高路由器性能。

请使用 **access-list compiled** 命令查看 turbo ACL。以下是已编译 ACL 的示例。

```
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
```

定义标准 ACL 或扩展 ACL 之后，可使用 **global configuration** 命令进行编译。

```
!--- 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
```

show access-list compiled 命令可显示有关 ACL 的统计数据。

基于时间的分布式 ACL

基于时间的分布式 ACL 是在 Cisco IOS 软件版本 12.2.2.T 中引入的，目的是在启用了 VPN 的 7500 系列路由器上实施基于时间的 ACL。在引入基于时间的分布式 ACL 功能之前，Cisco 7500 系列路由器的板卡并不支持基于时间的 ACL。配置了基于时间的 ACL 后，它们便可像正常 ACL 一样运行。如果为板卡上的接口配置了基于时间的 ACL，则交换到接口中的数据包并不会通过板卡进行分布式交换，而是转发到路由处理器中进行处理。

基于时间的分布式 ACL 和基于时间的 ACL 语法相同，但添加了与路由处理器和板卡之间的处理器间通信 (IPC) 消息状态有关的命令。

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

接收 ACL

接收 ACL 旨在通过防止路由器的千兆路由处理器 (GRP) 收到不必要和潜在非法数据流来提高 Cisco 12000 路由器的安全性。接收 ACL 作为一个特殊的放弃被添加到 Cisco IOS 软件版本 12.0.21S2 的维护扼杀中，并集成到了 12.0(22)S 中。请参阅 GSR：接收访问控制列表以了解更多信息。

基础架构保护 ACL

基础架构 ACL 旨在通过明确允许仅授权的数据流传输至基础架构设备且同时允许所有其他中转数据流，将对基础架构的直接攻击的风险和有效性降至最低。有关基础架构 ACL 的详细信息，请参阅保护您的核心：基础架构保护访问控制列表以了解更多信息。

中转 ACL

由于中转 ACL 明确规定仅允许所需的数据流进入网络，因此它们可用于提高网络安全性。请参阅中转访问控制列表：在网络边缘执行过滤以了解更多信息。

相关信息

- RFC 1700
- RFC 1918
- 访问列表支持页面
- Cisco IOS 防火墙
- Cisco IOS 软件 - 支持资源
- 技术支持和文档 - Cisco Systems