Rev. 20180926.151007

(ICND1) 25. Standard ACLs
(ICND2) 16. Standard ACLs

ccnacookbook.com

BASICS

ACLs (Access Control Lists) are matched against packets traversing a router to identify packets for special processing, often for dropping.

Traffic Filtering—Once defined, an ACL can be applied to an interface in a specified direction.

Outbound ACLs don't affect traffic generated within the router (pings, routing protocol updates)!

ACL Numbering—All statements with the same ACL number are part of the same ACL and IOS magically maintains the order in which they were entered and executes each statement of an ACL in that order. Traditionally, the way to change one line was to start over completely.

ACL Type	Number Ranges	Can Match Packets Based On
Standard	1-99 1300 - 1999	IP Address—Source Only
Extended [Next Chapter]	100-199 2000 - 2699	IP Address—Source and Destination TCP/UDP Port Number—Source and Destination Other Things

ACL Logic—An ACL can have multiple statements (lines) within it, each representing a single comparison criteria. As IOS works its way down the lines of an ACL, it will immediately take action on the first line that matches and stop processing the rest of the list.

Implicit Deny—If no line matches, then an implicit (imaginary) "deny everything else" is acted on.

Standard ACL Syntax—choose one from each column:

Command	ACL Number	Action	Source IP Address	Logging

`access-list`	`1 - 99`
	`1300 - 1999`
		`permit`
		`deny`
			`host 10.0.0.1`
			`10.0.0.1`
			`10.0.0.0 0.0.0.255`
			`any`
				`log`

		`remark`	`say whatever you want`	

Wildcard—A 32-bit mask applied to the address, where for each bit, a 0 says the address must match and a 1 says we don't care. The "0.0.0.255" that follows "10.0.0.0" in the table above (column 4) is a wildcard mask. Wildcards overcome the limitations of subnet masks by allowing1s and 0s to be interleaved arbitrarily. For example, the address & wildcard combination "10.0.0.1 255.255.255.254" means "all odd numbered IP addresses" (the first 31 bits are 1 and mean "we don't care," while the last bit is 0, requiring that the last bit of the packet source address match the last bit of the given address, which is a 1 and therefore an odd number).

Wildcard Bit	Meaning
0	Must Match
1	Don't Care

If all you want is to simply express a subnet mask as a wildcard mask, subtract each octet of the subnet mask from 255 to get a wildcard mask.

MASK	WILDCARD
255.255.255.0	0.0.0.255

Interestingly, IOS might simplify the address mentioned in an ACL in response to the wildcard. If a wildcard octet is 255, that octet of the address can be simplified to 0. Also, the wildcards 0.0.0.0 and 255.255.255.255 can be expressed more clearly in an ACL by using the keywords "host" and "any." Finally, IOS actually drops the "host" keyword or "0.0.0.0" wildcard from a standard ACL in the running-config ("show run"). The following fragments are all equivalent:

ADDRESS + WILDCARD	KEYWORD
172.16.12.88 0.0.0.0	host 172.16.12.88
10.3.1.7 255.255.255.255	any

```
permit 72.163.4.161 0.0.0.0
permit host 72.163.4.161
permit 72.163.4.161
```

This last option is what a router will show in its running-config

Example ACL:

```
R6(config)# access-list 10 remark Local Traffic Only
R6(config)# access-list 10 deny host 192.168.3.22
```

Exceptions must be typed before a broader rule (see the next line down) because the router will take action and stop processing the list after the first match

```
R6(config)# access-list 10 permit 192.168.0.0 0.0.255.255
R6(config)# access-list 10 permit 172.16.0.0 0.0.15.255
```

This allows traffic with source addresses of 172.16.0.0 through 172.31.255.255 and illustrates a useful coincidence: adding the network address and the wildcard or an address range gives you the ending address of that range.

Logging—If the keyword "log" is tacked onto the end of an ACL statement, then IOS will place a message in the system log whenever a match occurs …if it has spare time between packets.

```
R6(config)# access-list 10 deny   192.168.3.22 log
```

ACL Placement—Standard ACLs are placed near the destination. Basically, their location specifies the destination they apply to and their contents specify the source. Their applied direction matches the flow toward the destination.

```
R6(config)# interface Gi0/1
R(config-if)# ip access-group 10 {in | out}
```

10 is the ACL number.

Note: Routers don't enforce access-lists against internally-generated traffic, so you can't effectively test an outbound access-list with a ping typed on the command line of the same router.

Placement on VTYs—An ACL can also be placed on a VTY interface (in the "management-plane" as opposed to the "data-plane") to limit which IP addresses are allowed to telnet/ssh into the device. [Chapter 26]

```
R(config)# line vty 0 15
R(config-line)# access-class 20 in
```

To keep the placement commands straight (access-class vs. access-group), I just remember that we do a lot more configuring of routers and switches in *class* than in real life, where routers and switches are mostly left alone.

Here's what the access-list looks like in the running config.

```
R6# show run | begin access-list
access-list 10 remark Local Traffic Only
access-list 10 deny   192.168.3.22 log
```

The router simplified what we typed. It could have been "host 192.168.3.22" or "192.168.3.22 0.0.0.0"

```
access-list 10 permit 192.168.0.0 0.0.255.255
access-list 10 permit 172.16.0.0 0.0.15.255
access-list 10 permit 10.0.0.0 0.255.255.255
```

To just see access-lists, you can type "show access lists" or "show ip access-lists." For us, the output will be the same since we're only creating access lists that filter IP packets.

```
R6# show access-lists
Standard IP access list 10
    10 deny   192.168.3.22
    20 permit 192.168.0.0, wildcard bits 0.0.255.255
    30 permit 172.16.0.0, wildcard bits 0.0.15.255
```

The numbers "10, 20, 30" in the left column are line numbers useful for editing (next chapter)

```
    40 permit 10.0.0.0, wildcard bits 0.255.255.255 (5 matches)
```

The number of historically matching packets are in parentheses. If the "log" keyword had been used on that line of the ACL, then the system log ("show log") would have the match details.

We can also limit the output to just one access-list

```
edge# show ip access-lists [number | name]
```

Named access lists are in the next chapter

We can also see what ACLs have been applied to an interface and in what direction. The "IP" in the command is important; we want the layer 3 command, not the layer 1&2 "show interfaces…"

```
R6# show ip interface gi0/1
GigabitEthernet0/1 is up, line protocol is up
  Internet address is 10.56.0.6/24
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Outgoing access list is not set
  Inbound  access list is 10
  …
```