

IPv4 ACLs (Access Control Lists)

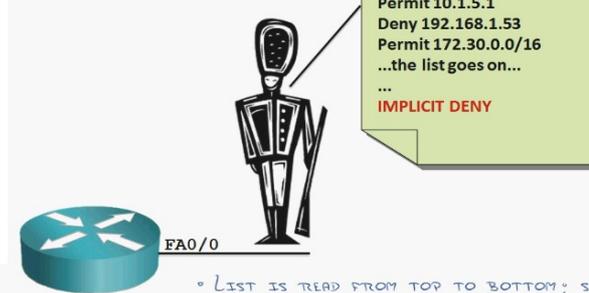
Standard ACL Syntax	Actions
<pre>! Legacy syntax access-list <number> {permit deny} <source> [log] ! Modern syntax ip access-list standard {<number> <name>} [<sequence>] {permit deny} <source> [log]</pre>	<pre>permit Allow matched packets deny Deny matched packets remark Record a configuration comment evaluate Evaluate a reflexive ACL</pre>

Extended ACL Syntax
<pre>! Legacy syntax access-list <number> {permit deny} <protocol> <source> [<ports>] <destination> [<ports>] [<options>] ! Modern syntax ip access-list extended {<number> <name>} [<sequence>] {permit deny} <protocol> <source> [<ports>] <destination> [<ports>] [<options>]</pre>

WHAT THEY CAN BE USED FOR:

- ACCESS CONTROL
- NAT
- QUALITY OF SERVICE
- DEMAND DIAL ROUTING
- POLICY ROUTING
- ROUTE FILTERING

USING ACLs FOR SECURITY



STANDARD AND EXTENDED ACCESS LISTS

- STANDARD
 - MATCHES BASED ON SOURCE ADDRESS
 - LOWER PROCESSOR UTILIZATION
 - AFFECT DEPENDS ON APPLICATION
- EXTENDED
 - MATCHES BASED ON SOURCE/DESTINATION ADDRESS, PROTOCOL, SOURCE/DESTINATION PORT NUMBER
 - HIGHER PROCESSOR UTILIZATION
 - SYNTAX TAKES SOME TIME TO LEARN
- REFLEXIVE (ESTABLISHED)
 - ALLOWS RETURN TRAFFIC FOR INTERNAL REQUESTS

- LIST IS READ FROM TOP TO BOTTOM; STOPS AT FIRST MATCH
- INVISIBLE IMPLICIT DENY AT THE BOTTOM
- ACL IS APPLIED TO AN INTERFACE INBOUND OR OUTBOUND
- OTHER POSSIBILITIES: ACL USED FOR NAT, QoS, VPN

ACL Numbers	TCP Options
1-99 IP standard	ack Match ACK flag
1300-1999 IP extended	fin Match FIN flag
100-199 IP extended	psh Match PSH flag
2000-2699 Protocol	rst Match RST flag
200-299 DECnet	syn Match SYN flag
300-399 XNS	urg Match URG flag
400-499 Extended XNS	established Match packets in an established session
500-599 Appletalk	Log Log ACL entry matches
600-699 Ethernet MAC	Log-input Log matches including ingress interface and source MAC address
700-799 IPX standard	
800-899 IPX extended	
900-999 IPX SAP	
1000-1099 MAC extended	
1100-1199 IPX summary	
1200-1299	

Source/Destination Definitions	
any	Any address
host <address>	A single address
<network> <mask>	Any address matched by the wildcard mask
IP Options	
dscp <DSCP>	Match the specified IP DSCP
fragments	Check non-initial fragments
option <option>	Match the specified IP option
precedence {0-7}	Match the specified IP precedence
ttl <count>	Match the specified IP time to live (TTL)
TCP/UDP Port Definitions	
eq <port>	Equal to
lt <port>	Less than
gt <port>	Greater than
neq <port>	Not equal to
range <port> <port>	Matches a range of port numbers
Miscellaneous Options	
reflect <name>	Create a reflexive ACL entry
time-range <name>	Enable rule only during the given time range

- » Packet filtering mechanism
- » Can filter packets on the basis of Layer 3 and Layer 4 header
- » Should have at least one permit statement
- » Works in sequential order; statement with lower sequence is preferred and checked

- » Only one ACL can be applied per interface, per direction
- » Can be applied inbound and outbound
 - Inbound - before routing
 - Outbound - after routing
- » Implicit deny rule applied at the end of the sequence if nothing has been defined

1. There is a deny all at the end of the list
 2. Every entry in an ACL is ACE (Access-list Entry)
- (randomly generated port numbers on the PCs are called ephemeral ports)

Standard ACL

- » Filters traffic based on Layer 3 header
- » Source IP address is checked
- » ACL numbers range from 1 through 99
- » Should be applied nearest to destination
- » No intelligence of checking destination address and port numbers

Extended ACL

- » Filters traffic based on layer 3 and 4 header
- » Source and destination IP and port numbers are checked
- » ACL numbers range from 100 through 199
- » Should be applied nearest to source
- » Capable of transport header inspection

Standard Access List: Standard Access List close to the destination are best.

```
R1(config)#access-list 1 permit 10.0.0.0 0.255.255.255
```

```
R1(config)#access-list 1 deny host 10.0.0.1 log
```

```
R1(config)#access-list 1 permit any any ! (don't forget this as there is a default deny at the end)
```

Extended Access list: Extended Access lists closer to the source are best.

```
R1(config)#access-list 101 permit tcp 10.0.0.0 0.255.255.255 187.100.1.6 0.0.0.0 eq 20
```

```
!(187.100.1.6 0.0.0.0 is the same as host 187.100.1.6)
```

```
R1(config)#access-list 101 deny tcp any eq 22 host 10.0.0.1 range 22 23
```

```
R1(config)#access-list 101 permit ip any any dscp cs2
```

Apply this ACL to an interface:

```
R1(config)#interface Fa0/1
```

```
R1(config-if)#ip access-group 1 out
```

OR

```
R1(config)#interface Fa0/0
```

```
R1(config-if)#ip access-group 1 in
```

Named Access Lists:

```
R(config)#ip access-list extended MyACL
```

```
R(config-ext-nacl)#100 permit ip host 1.1.1.1 any
```

Edit and Insert Lines in Access Lists:

```
R(config)#ip access-list extended MyACL
```

```
R(config-ext-nacl)#no 500
```

```
R(config-ext-nacl)#500 permit ip any host 5.5.5.5
```

```
R(config-ext-nacl)#510 permit ip any host 6.6.6.6
```

Time-based ACLs:

```
R(config)#time-range TR_WORKDAYS
```

```
R(config-time-range)#periodic weekdays 08:00 to 19:00
```

```
!(Don't configure NTP unless mentioned in the LAB)
```

```
R(config)#ip access-list extended 100
```

```
R(config-ext-nacl)#27 permit tcp any any eq www time-range TR_WORKDAYS
```

Access List Logging

- » Log message can be generated on ACL match
 - log vs. log-input
 - Generated as syslog level "informational"
 - Causes packets to be process switched
- » ACL Logging rate-limiting
 - ip access-list logging interval
 - ip access-list log-update threshold
 - logging rate-limit
- » ACL Syslog Correlation Tags
 - log [cookie]
 - ip access-list logging hash-generation

Named ACL

- » Individual statements can be edited, unlike numbered ACLs
- » Can be used with naming convention
- » Use of name instead of number makes management easier
- » More flexible than numbered ACLs

Time Based ACLs

- » Used to activate ACL entry based on clock
- » Defined as time-range [name]
 - Absolute
 - At one specific time period
 - Periodic
 - At one or more recurring time periods
- » Potential Applications
 - Time based traffic filter
 - Time based QoS

Restricting Telnet access with an Access-list:

```
R(config)#access-list 50 permit host 10.20.2.100
```

```
R(config)#line vty 0 15
```

```
R(config-line)#access-class 50 in
```

IPv4 ACLs: WHAT CAN GO WRONG?

- ACCESS LISTS APPLIED THE WRONG DIRECTION (VIA ACCESS GROUP OR MISSED LOGIC)
- FORGOTTEN IMPLICIT DENY (ALWAYS MANUALLY ADD IT)
- EARLIER MATCH, UNEXPECTED RESULTS
- PORT NUMBER IN THE WRONG PLACE



Verification and TSHOOT Commands:



(In the lab they could simply say to permit echo requests between two router (e.g. R3 and R1), but you need to be careful if you are running any routing protocols (eigrp or rip) to be permitted too, even if they haven't explicitly mentioned. Always verify by using log or log-input at the end of the ACLs and with debug commands).

```
R2(config)#access-list 100 permit icmp host 3.3.3.3 host 1.1.1.1 echo
```

```
R2(config)#access-list 100 permit eigrp any any
```

```
R2(config)#access-list 100 deny ip any any log
```

```
R2(config)#access-list 100 deny ip any any log-input
```

```
R(config)#ip access-list logging interval 500 !(in ms. This is the interval time it takes to send it to syslog)
```

```
R(config)#ip icmp rate-limit unreachable 100 !(in ms)
```

!(problem with this is that router will have to generate a huge amount of unreachable messages if someone pings hugely)

```
sh access-lists
```

```
sh ip access-lists
```

```
sh access-lists 1
```

```
sh ip access-lists 1
```

```
sh access-lists ACL_NAME1
```

```
sh ip access-lists ACL_NAME1
```

```
sh ip access-lists interface fa0/0
```

```
sh ip int fa0/0
```

```
sh time-range
```

```
sh time-range ACL_NAME1
```

```
sh run | inc access-list
```