CCNA 200-301, Volume 2

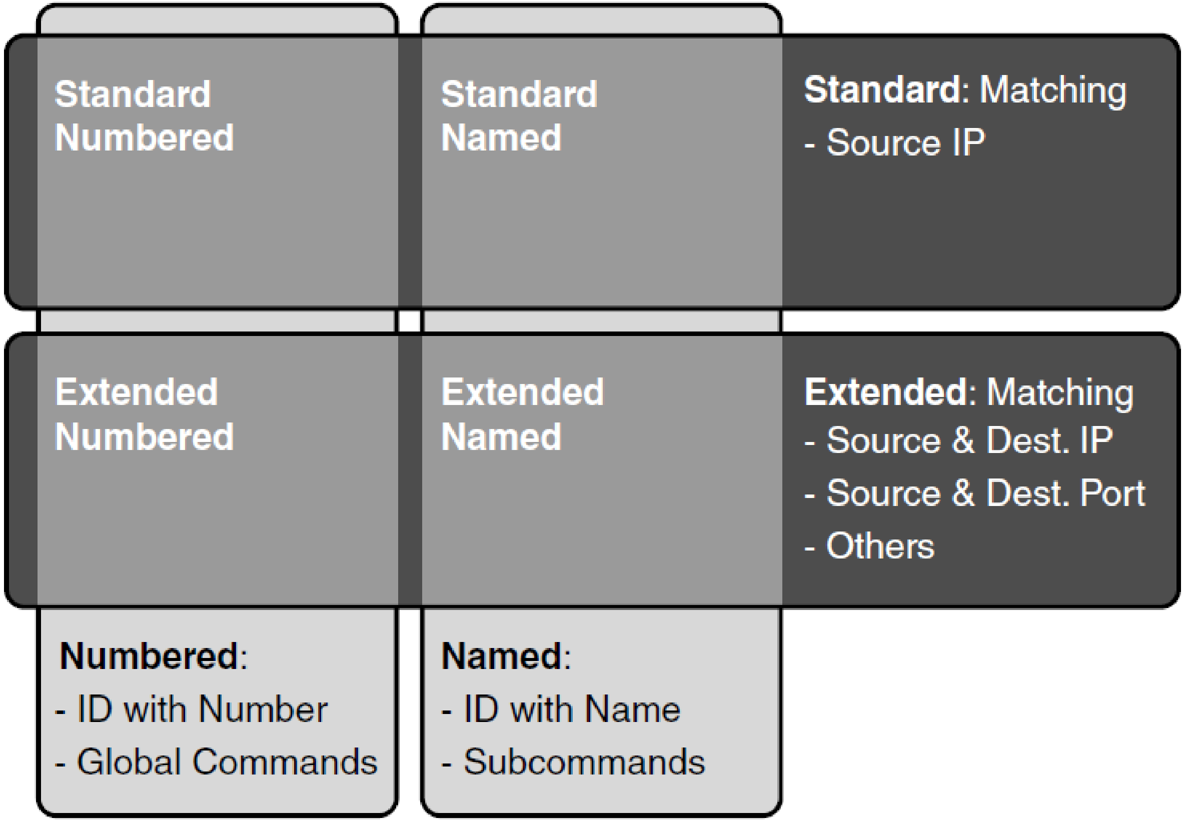
Chapter 3

Advanced IPv4 Access Control Lists

Objectives

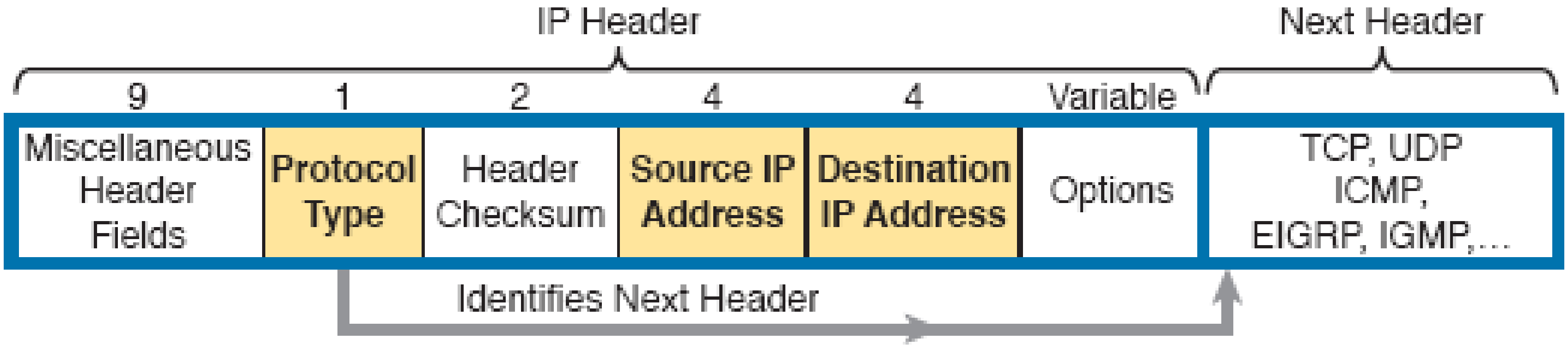
- Configure and verify access control lists

Comparisons of IP ACL Types

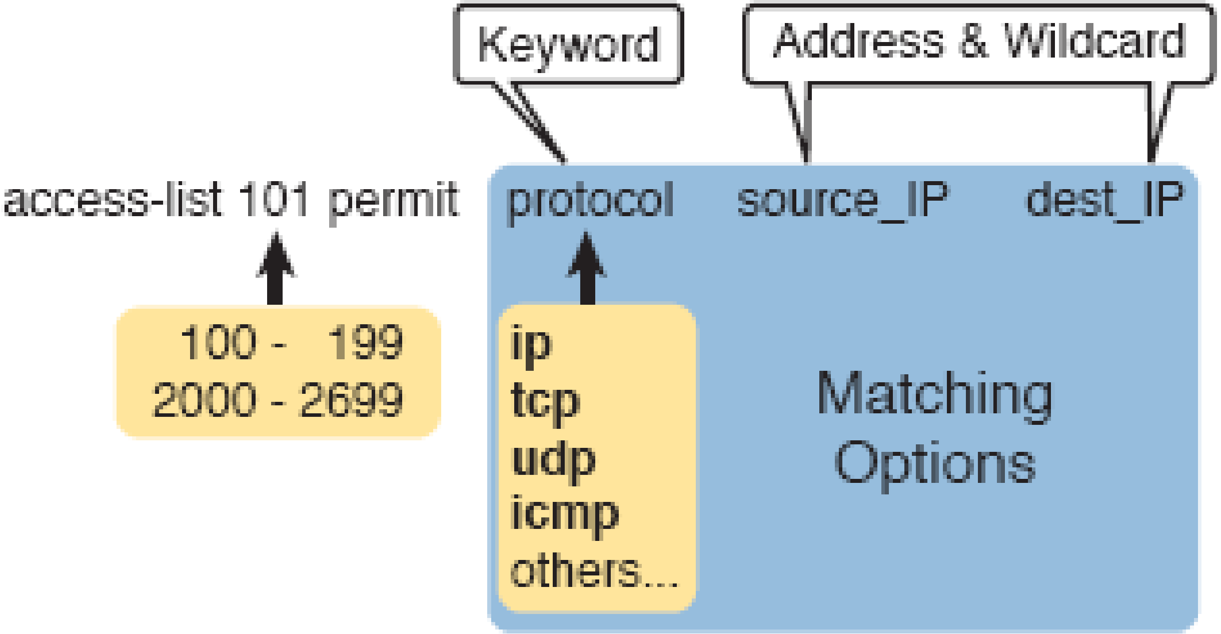


Standard Numbered	**Standard Named**	**Standard**: Matching - Source IP
Extended Numbered	**Extended Named**	**Extended**: Matching - Source & Dest. IP - Source & Dest. Port - Others
Numbered: - ID with Number - Global Commands	**Named**: - ID with Name - Subcommands	

IP Header, with Focus on Required Fields in Extended IP ACLs



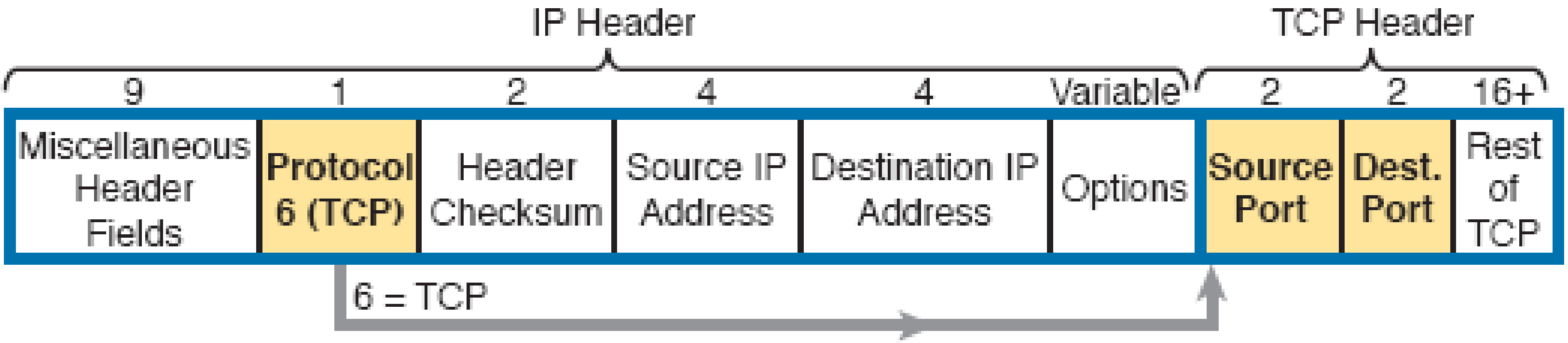
Extended ACL Syntax, with Required Fields



Extended **access-list** Commands and Logic Explanations

access-list Statement	**What it Matches**
access-list 101 deny tcp any any	Any IP packet that has a TCP header
access-list 101 deny udp any any	Any IP packet that has a UDP header
access-list 101 deny icmp any any	Any IP packet that has an ICMP header
access-list 101 deny ip host 1.1.1.1 host 2.2.2.2	All IP packets from host 1.1.1.1 going to host 2.2.2.2, regardless of the header after the IP header
access-list 101 deny udp 1.1.1.0 0.0.0.255 any	All IP packets that have a UDP header following the IP header, from subnet 1.1.1.0/24, and going to any destination

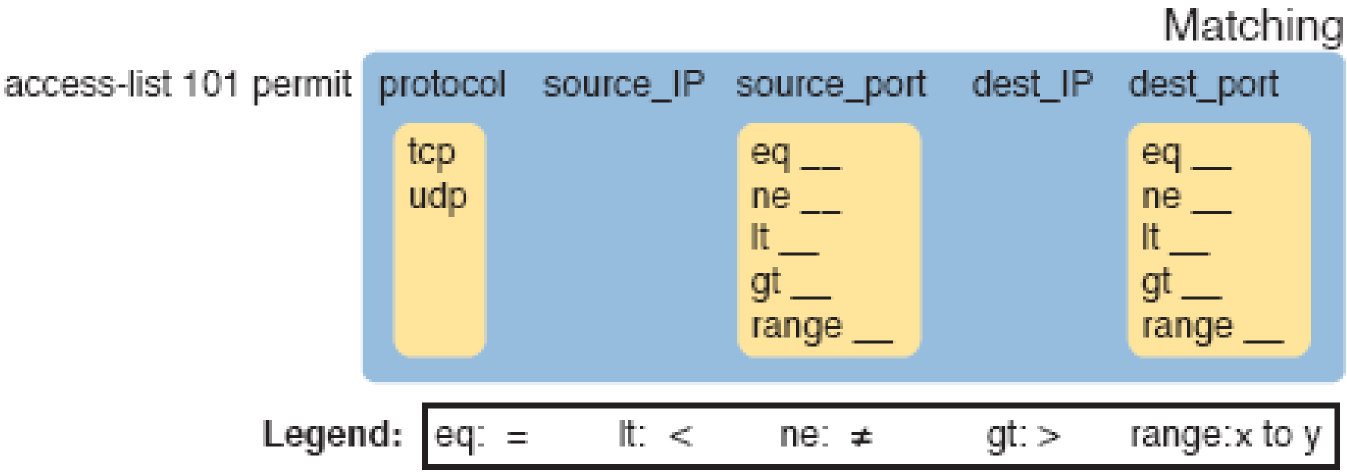
IP Header, Followed by a TCP Header and Port Number Fields



IP Header						TCP Header		
9	1	2	4	4	Variable	2	2	16+
Miscellaneous Header Fields	**Protocol 6 (TCP)**	Header Checksum	Source IP Address	Destination IP Address	Options	**Source Port**	**Dest. Port**	Rest of TCP

6 = TCP

Extended ACL Syntax with TCP and UDP Port Numbers Enabled



Matching

access-list 101 permit protocol source_IP source_port dest_IP dest_port

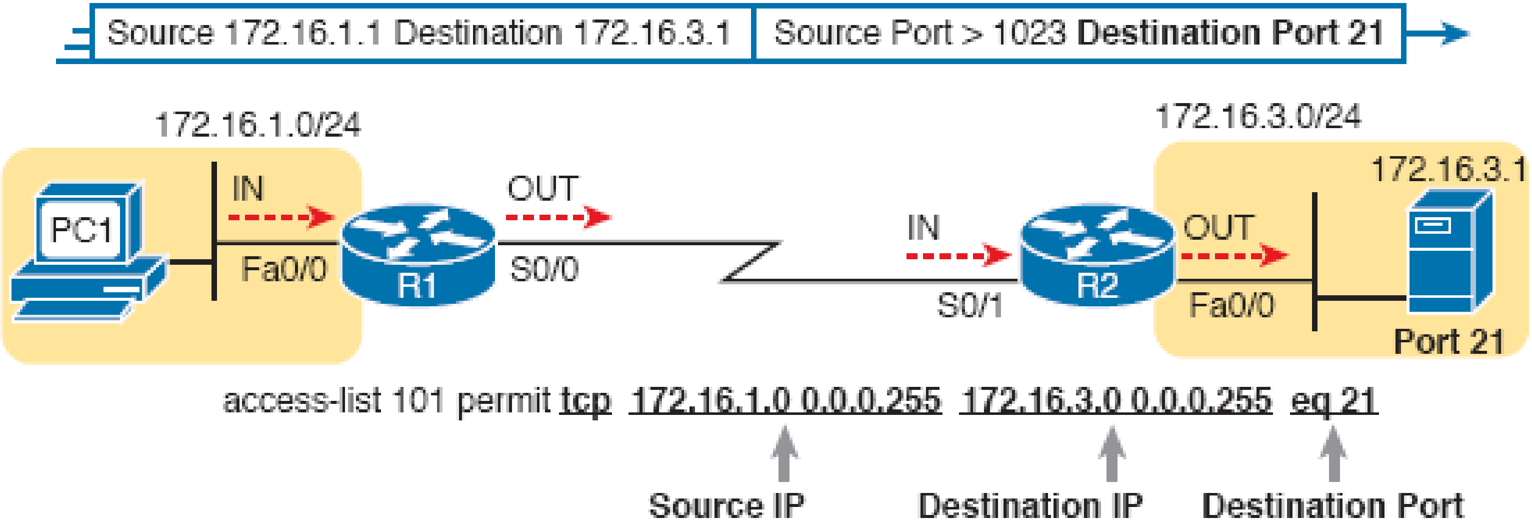
tcp
udp

eq __
ne __
lt __
gt __
range __

eq __
ne __
lt __
gt __
range __

Legend: eq: = lt: < ne: ≠ gt: > range:x to y

Filtering Packets Based on Destination Port



Source 172.16.1.1 Destination 172.16.3.1 | Source Port > 1023 **Destination Port 21**

172.16.1.0/24

PC1

IN

Fa0/0

R1

OUT

S0/0

IN

S0/1

R2

OUT

Fa0/0

172.16.3.0/24

172.16.3.1

Port 21

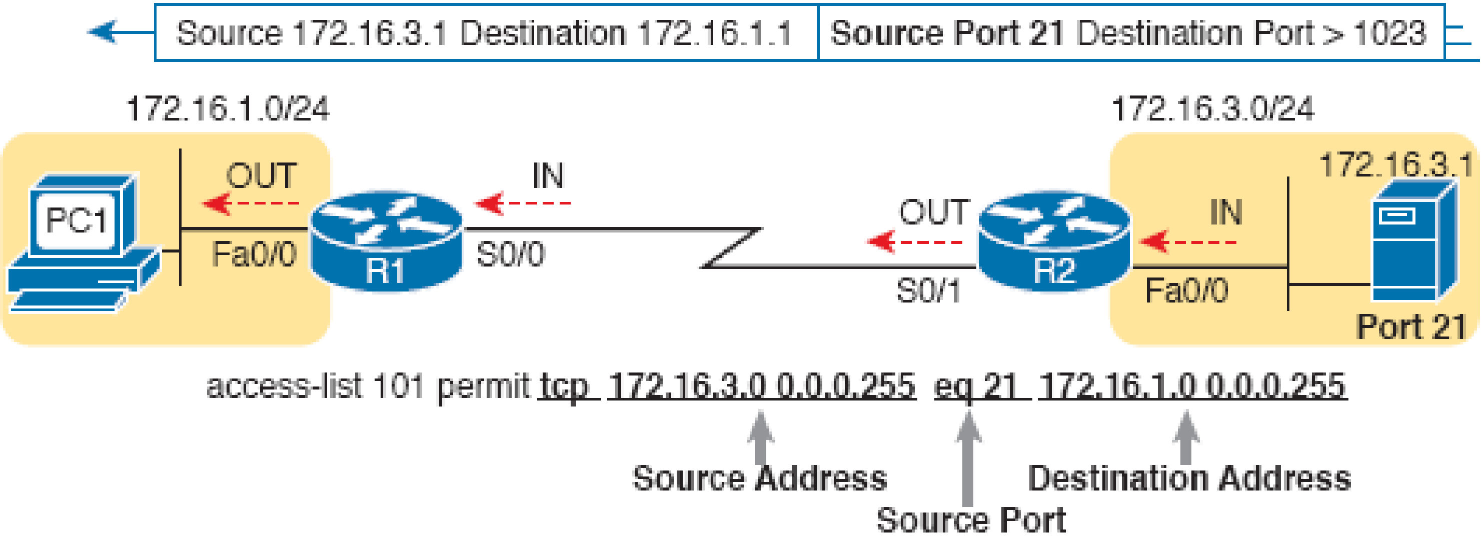
access-list 101 permit tcp 172.16.1.0 0.0.0.255 172.16.3.0 0.0.0.255 eq 21

Source IP

Destination IP

Destination Port

Filtering Packets Based on Source Port



Source 172.16.3.1 Destination 172.16.1.1
Source Port 21 Destination Port > 1023
172.16.1.0/24
172.16.3.0/24
PC1
OUT
Fa0/0
R1
IN
S0/0
OUT
S0/1
R2
IN
Fa0/0
172.16.3.1
Port 21
access-list 101 permit tcp 172.16.3.0 0.0.0.255 eq 21 172.16.1.0 0.0.0.255
Source Address
Source Port
Destination Address

Popular Applications and Their Well-Known Port Numbers

Port Number(s)	**Protocol**	**Application**	**access-list** Command Keyword
20	TCP	FTP Data	**ftp-data**
21	TCP	FTP control	**ftp**
22	TCP	SSH	--
23	TCP	Telnet	**telnet**
25	TCP	SMTP	**smtp**
53	UDP, TCP	DNS	**domain**
67	UDP	DHCP Server	**bootps**
68	UDP	DHCP Client	**bootpc**

Popular Applications and Their Well-Known Port Numbers (continued)

Port Number(s)	**Protocol**	**Application**	**access-list** Command Keyword
69	UDP	TFTP	**tftp**
80	TCP	HTTP (WWW)	**www**
110	TCP	POP3	**pop3**
161	UDP	SNMP	**snmp**
443	TCP	SSL	**--**
514	UDP	Syslog	**--**
16,384-32,767	UDP	RTP (Voice, Video)	**--**

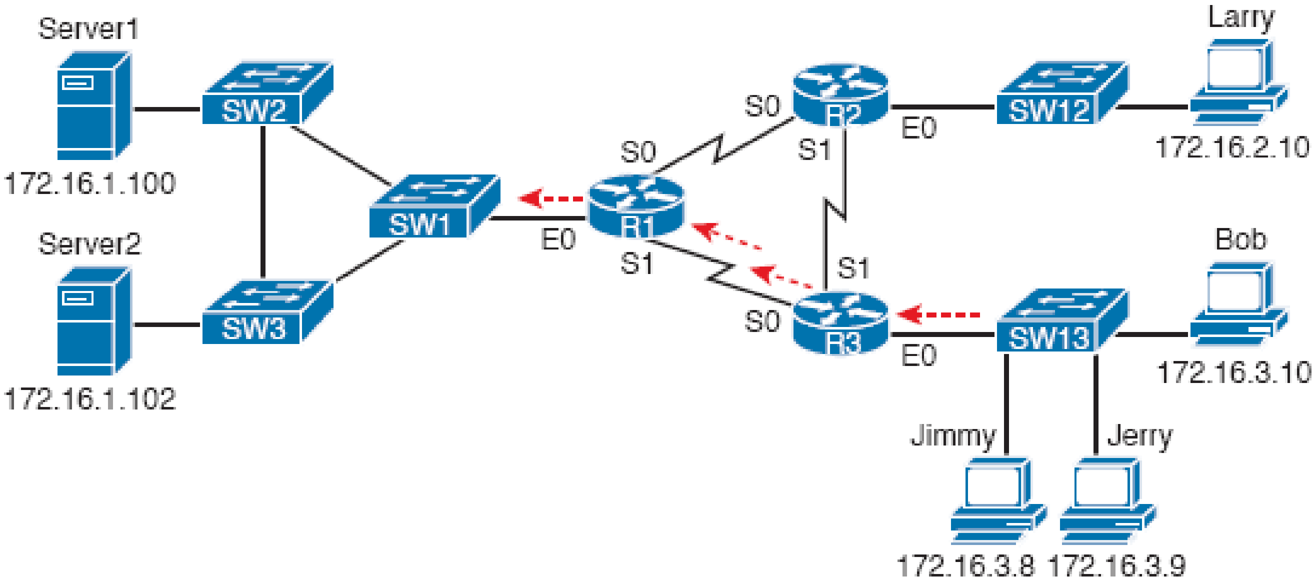
Extended **access-list** Command Examples and Logic Explanations

access-list Statement	What It Matches
access-list 101 deny tcp any gt 49151 host 10.1.1.1 eq 23	Packets with a TCP header, any source IP address, with a source port greater than (gt) 1023, a destination IP address of exactly 10.1.1.1, and a destination port equal to (eq) 23.
access-list 101 deny tcp any host 10.1.1.1 eq 23	The same as the preceding example, but any source port matches, because that parameter is omitted in this case.
access-list 101 deny tcp any host 10.1.1.1 eq telnet	The same as the preceding example. The **telnet keyword is** used instead of port 23.
access-list 101 deny udp 1.0.0.0 0.255.255.255 lt 1023 any	A packet with a source in network 1.0.0.0/8, using UDP with a source port less than (lt) 1023, with any destination IP address.

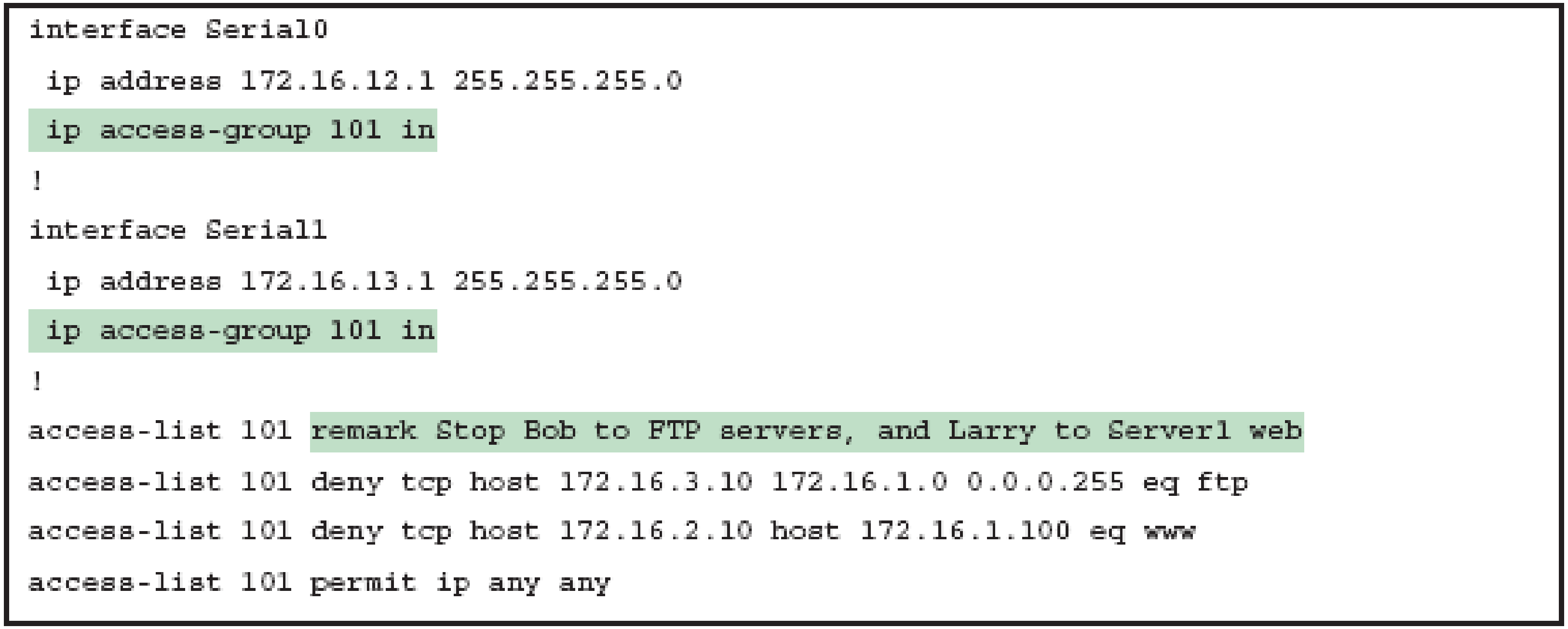
Extended IP Access List Configuration Commands

Command	Configuration Mode and Description
access-list ***access-list-number {deny*** **|** ***permit} protocol*** *source source-wildcard destination destination-wildcard* [**log | log-input]**	Global command for extended numbered **access lists. Use a** number between 100 and 199 or 2000 and 2699, inclusive.
access-list ***access-list-number {deny*** | ***permit} {tcp*** | ***udp}*** *source source-wildcard [operator [port]] destination destination-wildcard [operator [port]]* ***[established] [log]***	A version of the **access-list** command with parameters specific to TCP and/or UDP.

Network Diagram for Extended Access List Example 1

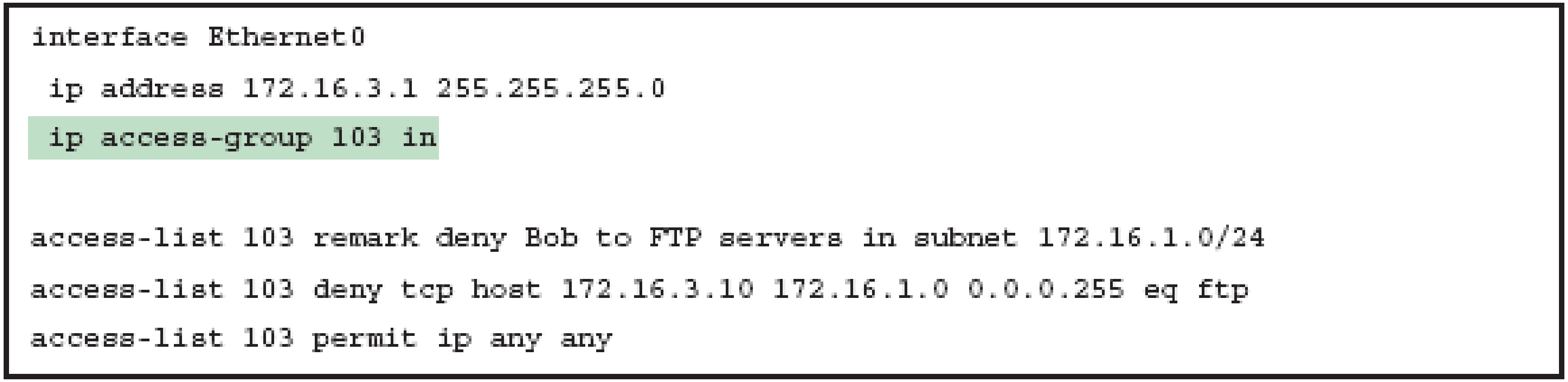


R1’s Extended Access List: Example 1



```
interface Serial0
 ip address 172.16.12.1 255.255.255.0
 ip access-group 101 in
!
interface Serial1
 ip address 172.16.13.1 255.255.255.0
 ip access-group 101 in
!
access-list 101 remark Stop Bob to FTP servers, and Larry to Server1 web
access-list 101 deny tcp host 172.16.3.10 172.16.1.0 0.0.0.255 eq ftp
access-list 101 deny tcp host 172.16.2.10 host 172.16.1.100 eq www
access-list 101 permit ip any any
```

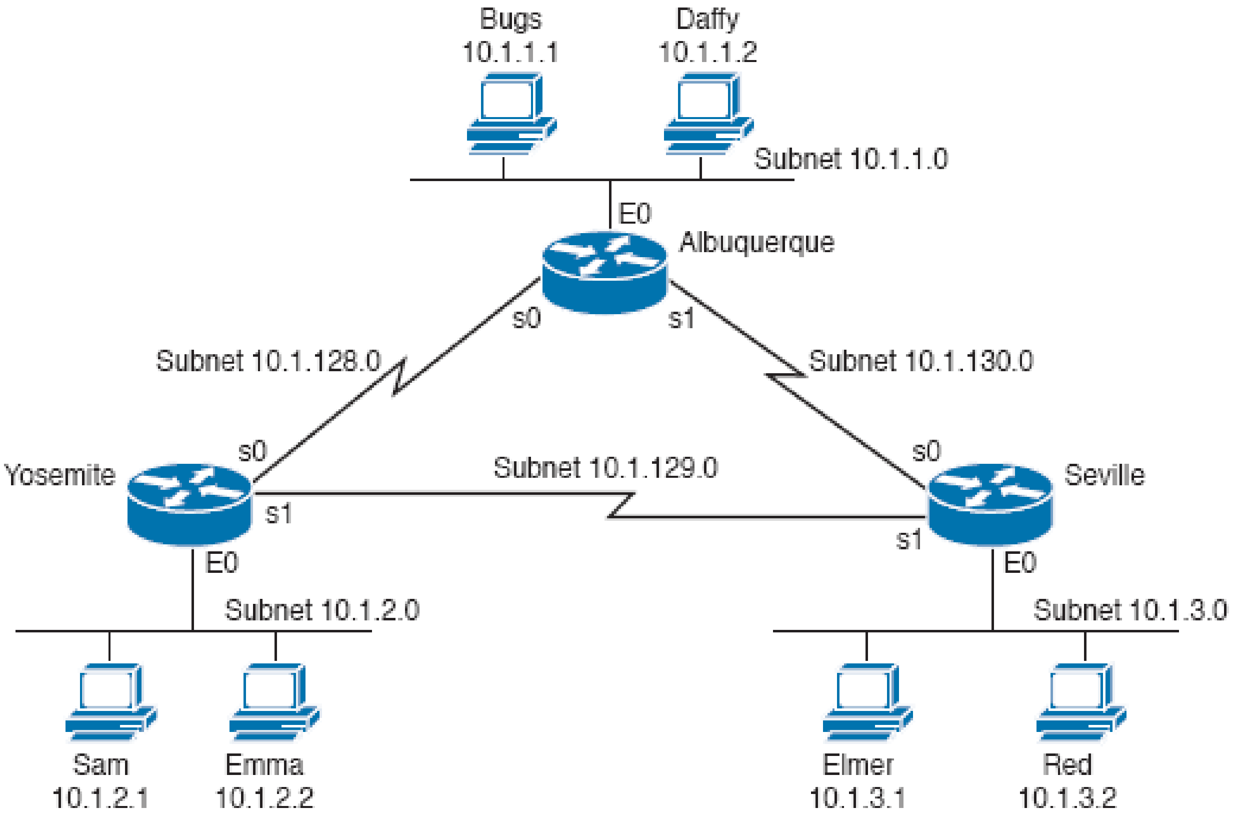
R3’s Extended Access List Stopping Bob from Reaching FTP Servers Near R1



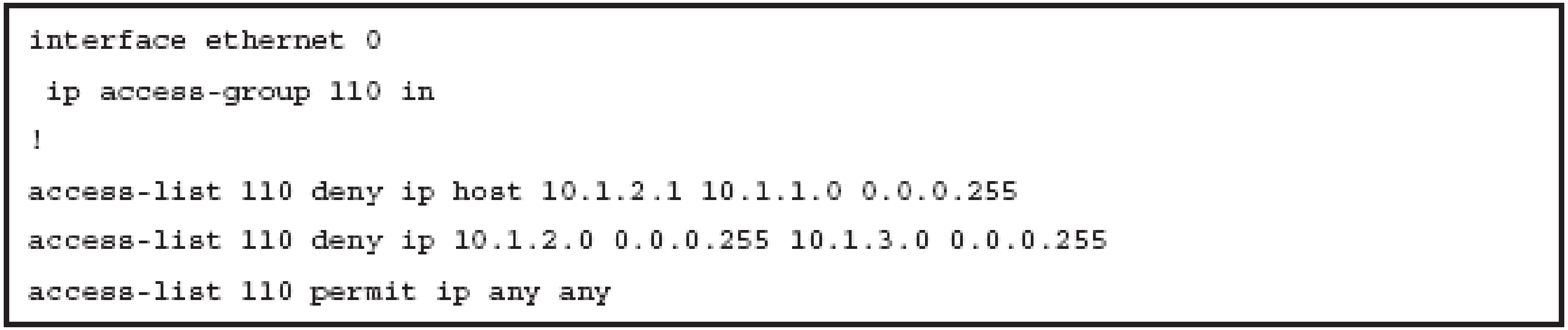
```
interface Ethernet0
 ip address 172.16.3.1 255.255.255.0
 ip access-group 103 in

access-list 103 remark deny Bob to FTP servers in subnet 172.16.1.0/24
access-list 103 deny tcp host 172.16.3.10 172.16.1.0 0.0.0.255 eq ftp
access-list 103 permit ip any any
```

Network Diagram for Extended Access List Example 2



Yosemite Configuration for Extended Access List Example 2



```
interface ethernet 0
 ip access-group 110 in
!
access-list 110 deny ip host 10.1.2.1 10.1.1.0 0.0.0.255
access-list 110 deny ip 10.1.2.0 0.0.0.255 10.1.3.0 0.0.0.255
access-list 110 permit ip any any
```

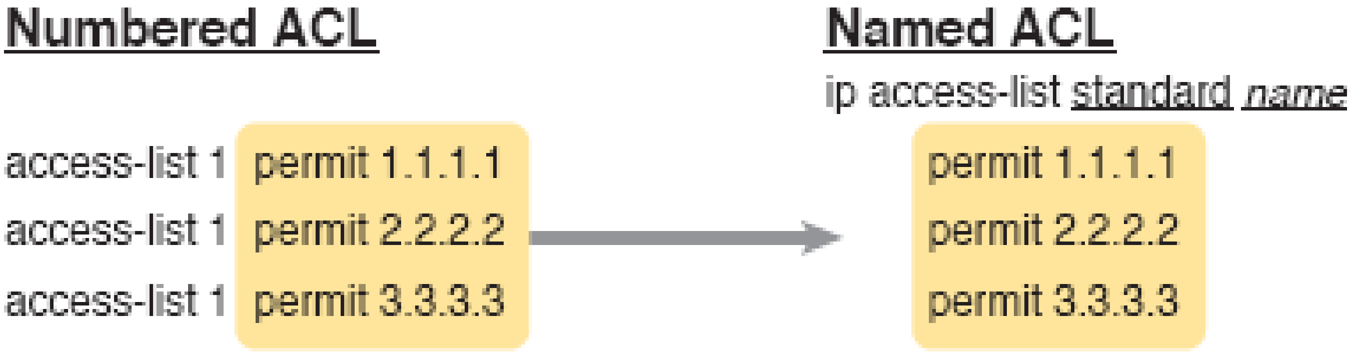
Building One-Line Extended ACLs: Practice

Problem	Criteria
1	From web client 10.1.1.1, sent to a web server in subnet 10.1.2.0/24.
2	From Telnet client 172.16.4.3/25, sent to a Telnet server in subnet 172.16.3.0/25. Match all hosts in the client's subnet as well.
3	ICMP messages from the subnet in which 192.168.7.200/26 resides to all hosts in the subnet where 192.168.7.14/29 resides.
4	From web server 10.2.3.4/23's subnet to clients in the same subnet as host 10.4.5.6/22.
5	From Telnet server 172.20.1.0/24's subnet, sent to any host in the same subnet as host 172.20.44.1/23.

Building One-Line Extended ACLs: Practice (continued)

Problem	Criteria
6	From web client 192.168.99.99/28, sent to a web server in subnet 192.168.176.0/28. Match all hosts in the client's subnet as well.
7	ICMP messages from the subnet in which 10.55.66.77/25 resides to all hosts in the subnet where 10.66.55.44/26 resides.
8	Any and every IPv4 packet.

Named ACL vs. Numbered ACL Configuration



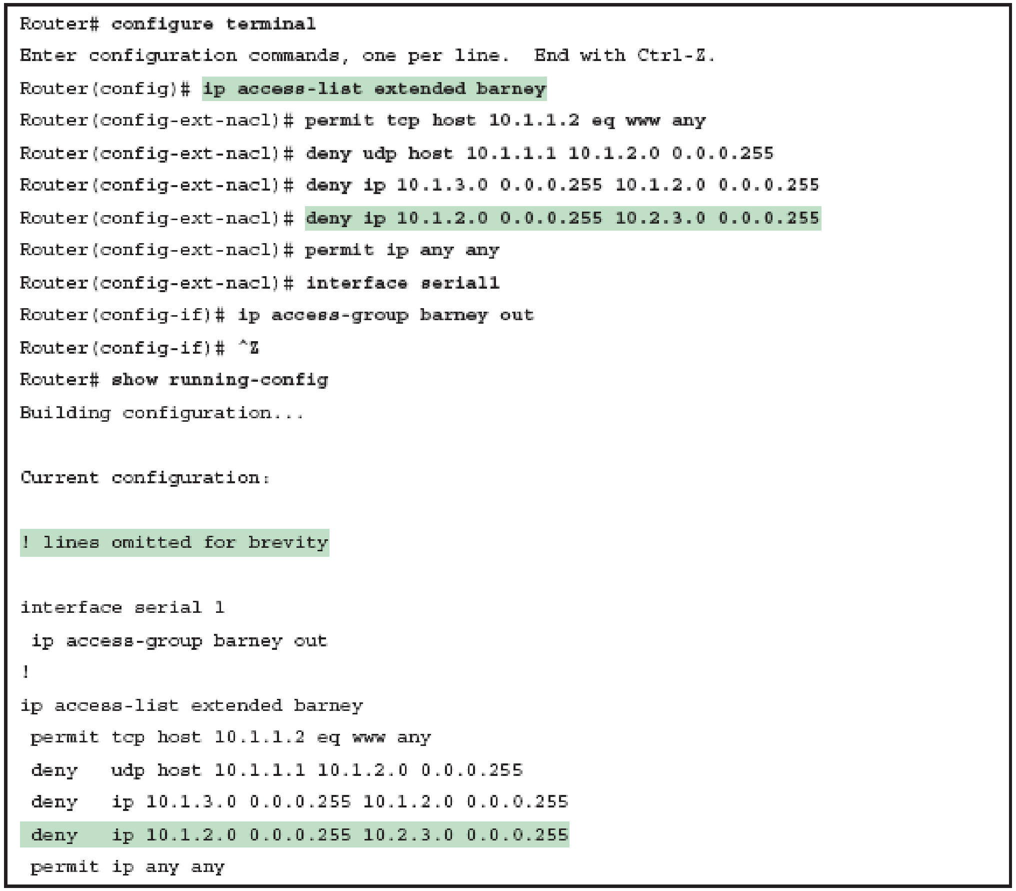
Numbered ACL

```
access-list 1 permit 1.1.1.1
access-list 1 permit 2.2.2.2
access-list 1 permit 3.3.3.3
```

Named ACL

```
ip access-list standard name
    permit 1.1.1.1
    permit 2.2.2.2
    permit 3.3.3.3
```

Named Access List Configuration



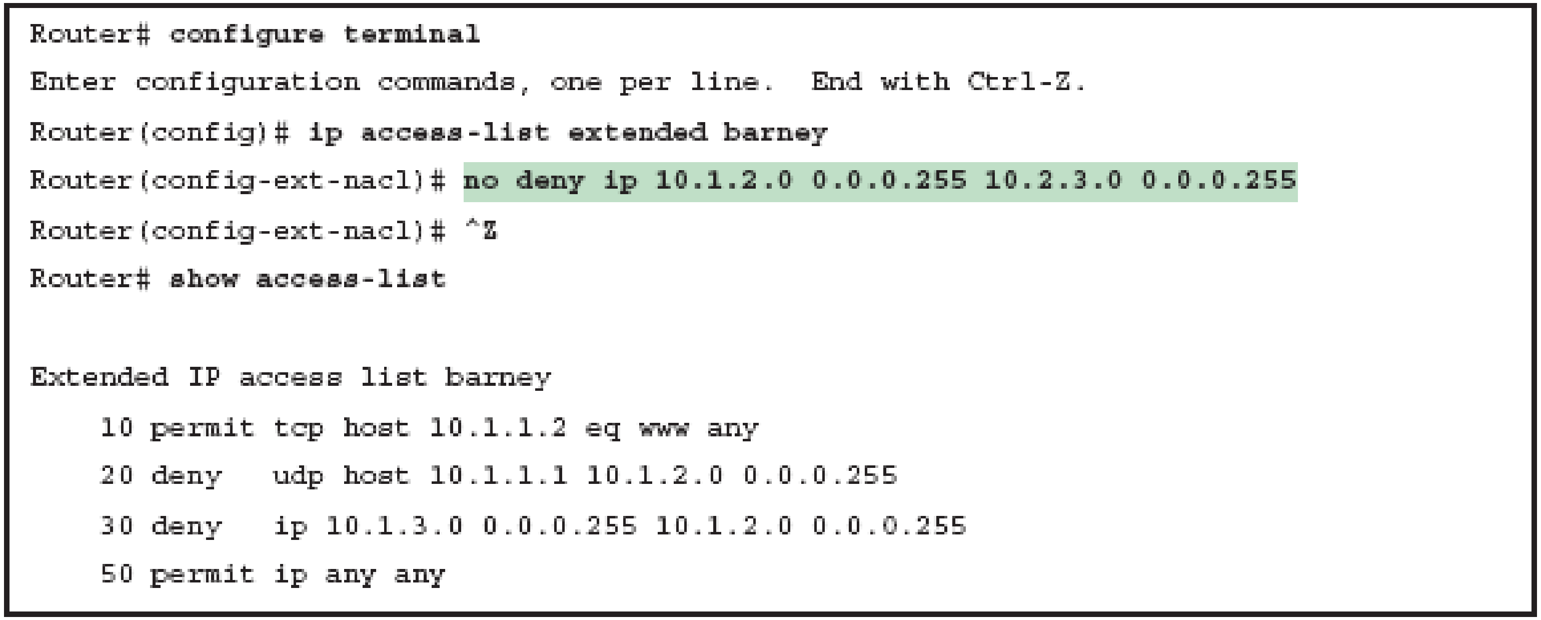
```
Router# configure terminal
Enter configuration commands, one per line.  End with Ctrl-Z.
Router(config)# ip access-list extended barney
Router(config-ext-nacl)# permit tcp host 10.1.1.2 eq www any
Router(config-ext-nacl)# deny udp host 10.1.1.1 10.1.2.0 0.0.0.255
Router(config-ext-nacl)# deny ip 10.1.3.0 0.0.0.255 10.1.2.0 0.0.0.255
Router(config-ext-nacl)# deny ip 10.1.2.0 0.0.0.255 10.2.3.0 0.0.0.255
Router(config-ext-nacl)# permit ip any any
Router(config-ext-nacl)# interface serial1
Router(config-if)# ip access-group barney out
Router(config-if)# ^Z
Router# show running-config
Building configuration...

Current configuration:

! lines omitted for brevity

interface serial 1
 ip access-group barney out
!
ip access-list extended barney
 permit tcp host 10.1.1.2 eq www any
 deny   udp host 10.1.1.1 10.1.2.0 0.0.0.255
 deny   ip 10.1.3.0 0.0.0.255 10.1.2.0 0.0.0.255
 deny   ip 10.1.2.0 0.0.0.255 10.2.3.0 0.0.0.255
 permit ip any any
```

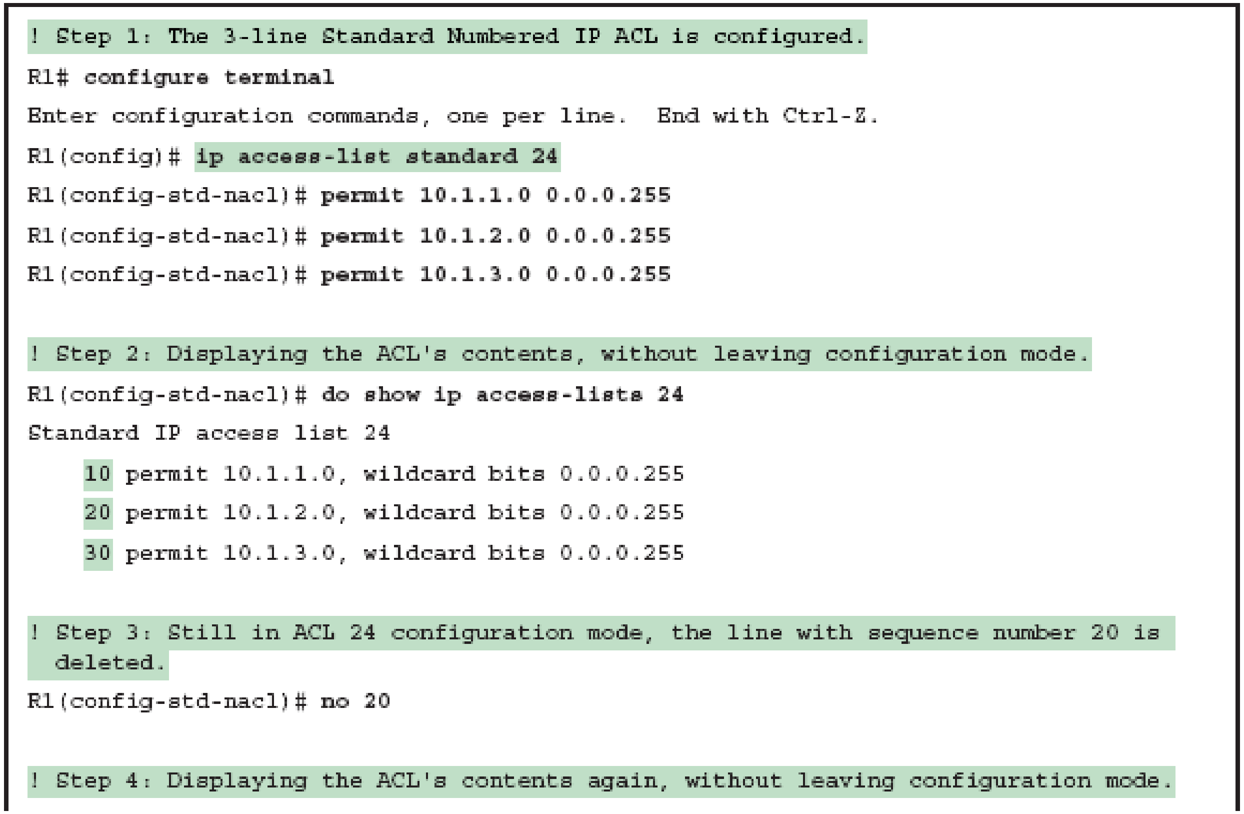
Removing One Command from a Named ACL



```
Router# configure terminal
Enter configuration commands, one per line.  End with Ctrl-Z.
Router(config)# ip access-list extended barney
Router(config-ext-nacl)# no deny ip 10.1.2.0 0.0.0.255 10.2.3.0 0.0.0.255
Router(config-ext-nacl)# ^Z
Router# show access-list

Extended IP access list barney
    10 permit tcp host 10.1.1.2 eq www any
    20 deny   udp host 10.1.1.1 10.1.2.0 0.0.0.255
    30 deny   ip 10.1.3.0 0.0.0.255 10.1.2.0 0.0.0.255
    50 permit ip any any
```

Editing ACLs Using Sequence Numbers



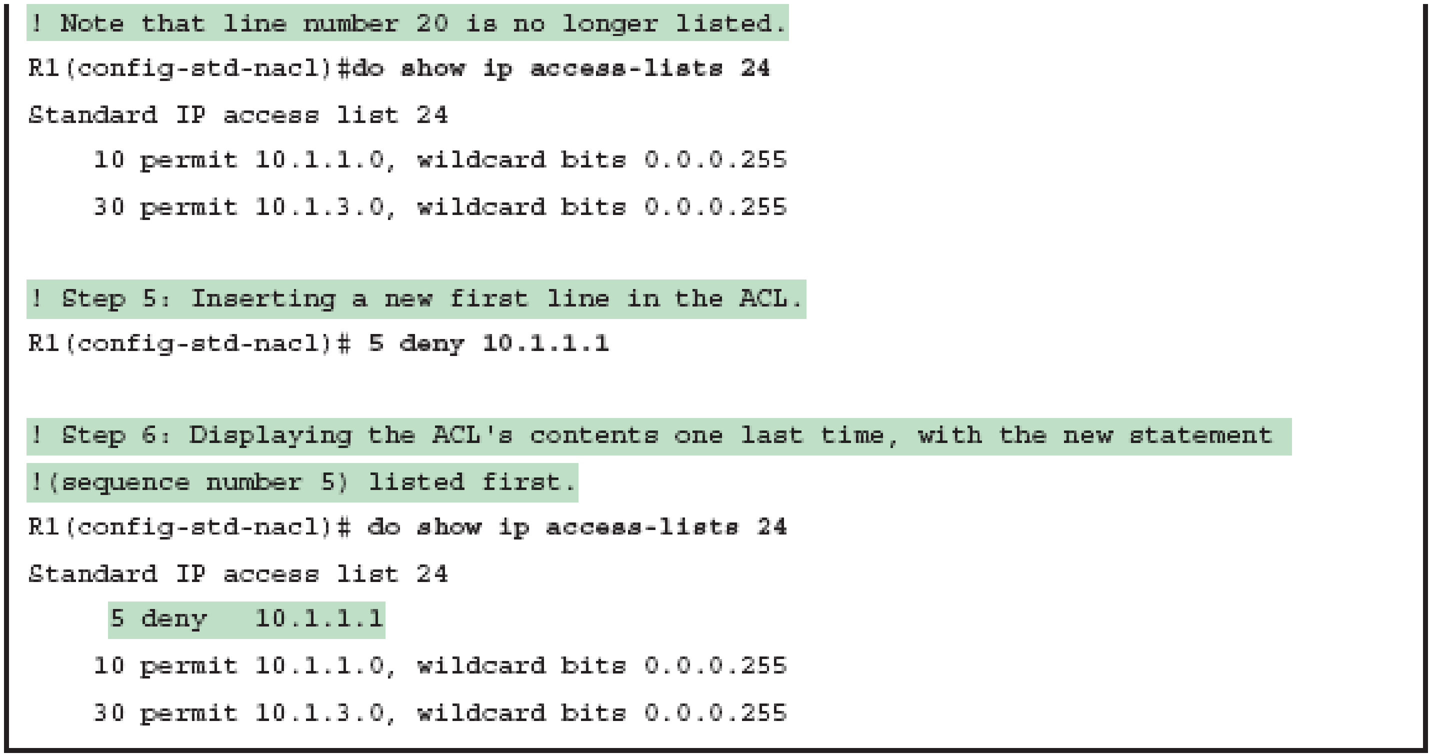
```
! Step 1: The 3-line Standard Numbered IP ACL is configured.
R1# configure terminal
Enter configuration commands, one per line.  End with Ctrl-Z.
R1(config)# ip access-list standard 24
R1(config-std-nacl)# permit 10.1.1.0 0.0.0.255
R1(config-std-nacl)# permit 10.1.2.0 0.0.0.255
R1(config-std-nacl)# permit 10.1.3.0 0.0.0.255

! Step 2: Displaying the ACL's contents, without leaving configuration mode.
R1(config-std-nacl)# do show ip access-lists 24
Standard IP access list 24
    10 permit 10.1.1.0, wildcard bits 0.0.0.255
    20 permit 10.1.2.0, wildcard bits 0.0.0.255
    30 permit 10.1.3.0, wildcard bits 0.0.0.255

! Step 3: Still in ACL 24 configuration mode, the line with sequence number 20 is
  deleted.
R1(config-std-nacl)# no 20

! Step 4: Displaying the ACL's contents again, without leaving configuration mode.
```

Editing ACLs Using Sequence Numbers (continued)

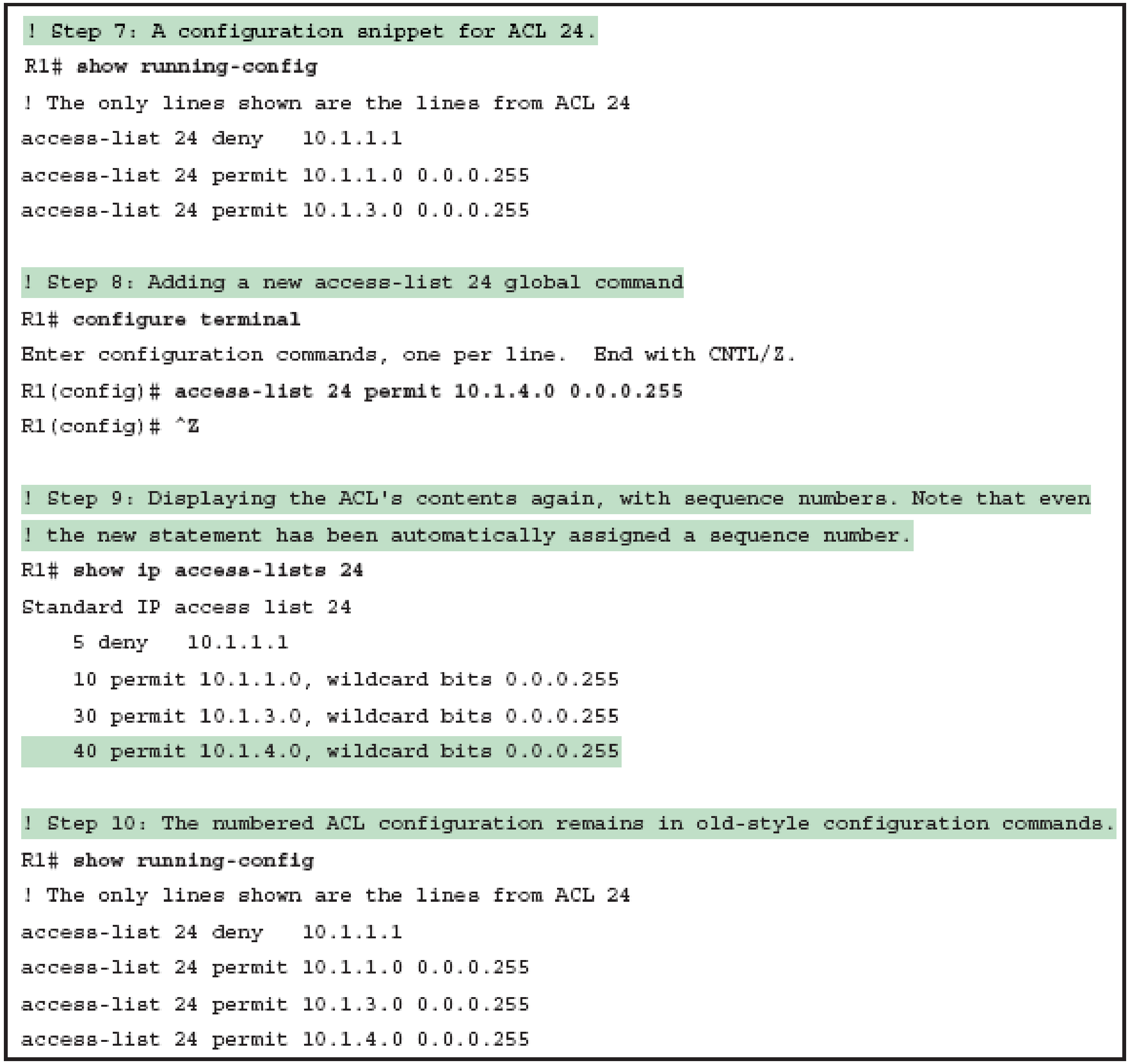


```
! Note that line number 20 is no longer listed.
R1(config-std-nacl)#do show ip access-lists 24
Standard IP access list 24
    10 permit 10.1.1.0, wildcard bits 0.0.0.255
    30 permit 10.1.3.0, wildcard bits 0.0.0.255

! Step 5: Inserting a new first line in the ACL.
R1(config-std-nacl)# 5 deny 10.1.1.1

! Step 6: Displaying the ACL's contents one last time, with the new statement
!(sequence number 5) listed first.
R1(config-std-nacl)# do show ip access-lists 24
Standard IP access list 24
     5 deny   10.1.1.1
    10 permit 10.1.1.0, wildcard bits 0.0.0.255
    30 permit 10.1.3.0, wildcard bits 0.0.0.255
```

Adding To and Displaying a Numbered ACL Configuration



```
! Step 7: A configuration snippet for ACL 24.
R1# show running-config
! The only lines shown are the lines from ACL 24
access-list 24 deny   10.1.1.1
access-list 24 permit 10.1.1.0 0.0.0.255
access-list 24 permit 10.1.3.0 0.0.0.255

! Step 8: Adding a new access-list 24 global command
R1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)# access-list 24 permit 10.1.4.0 0.0.0.255
R1(config)# ^Z

! Step 9: Displaying the ACL's contents again, with sequence numbers. Note that even
! the new statement has been automatically assigned a sequence number.
R1# show ip access-lists 24
Standard IP access list 24
    5 deny   10.1.1.1
    10 permit 10.1.1.0, wildcard bits 0.0.0.255
    30 permit 10.1.3.0, wildcard bits 0.0.0.255
    40 permit 10.1.4.0, wildcard bits 0.0.0.255

! Step 10: The numbered ACL configuration remains in old-style configuration commands.
R1# show running-config
! The only lines shown are the lines from ACL 24
access-list 24 deny   10.1.1.1
access-list 24 permit 10.1.1.0 0.0.0.255
access-list 24 permit 10.1.3.0 0.0.0.255
access-list 24 permit 10.1.4.0 0.0.0.255
```

General Recommendations for ACL Implementation

- Place extended ACLs as close as possible to the source of the packet.
- Place standard ACLs as close as possible to the destination of the packet.
- Place more specific statements early in the ACL.
- Disable an ACL from its interface (using the no ip access-group interface subcommand) before making changes to the ACL.