Access Lists

Workbook

Version 1.2

Instructor's Edition

Access-List Numbers

IP Standard	1	to	99
IP Extended	100	to	199
Ethernet Type Code	200	to	299
Ethernet Address	700	to	799
DECnet and Extended DECnet	300	to	399
XNS	400	to	499
Extended XNS	500	to	599
Appletalk	600	to	699
48-bit MAC Addresses	700	to	799
IPX Standard	800	to	899
IPX Extended	900	to	999
IPX SAP (service advertisement protocol)	1000	to	1099
IPX SAP SPX	1000	to	1099
Extended 48-bit MAC Addresses	1100	to	1199
IPX NLSP	1200	to	1299
IP Standard, expanded range	1300	to	1999
IP Extended, expanded range	2000	to	2699
SS7 (voice)	2700	to	2999
Standard Vines	1	to	100
Extended Vines	101	to	200
Simple Vines	201	to	300
Transparent bridging (protocol type)	200	to	299
Transparent bridging (vendor type)	700	to	799
Extended Transparent bridging	1100	to	1199
Source-route bridging (protocol type)	200	to	299
Source-route bridging (vendor type)	700	to	799

Produced by: Robb Jones
jonesr@careertech.net
Frederick County Career & Technology Center
Cisco Networking Academy
Frederick County Public Schools
Frederick, Maryland, USA

Special Thanks to Melvin Baker and Jim Dorsch
for taking the time to check this workbook for errors.

What are Access Control Lists?

ACLs...

...are a sequential list of instructions that tell a router which packets to permit or deny.

General Access Lists Information

Access Lists...

...are read sequentially.

...are set up so that as soon as the packet matches a statement it stops comparing and permits or denys the packet.

...need to be written to take care of the most abundant traffic first.

...must be configured on your router before you can deny packets.

...can be written for all supported routed protocols; but each routed protocol must have a different ACL for each interface.

...must be applied to an interface to work.

How routers use Access Lists

(Outbound Port - Default)

- The router checks to see if the packet is routable. If it is it looks up the route in its routing table.
- The router then checks for an ACL on that outbound interface.
- If there is no ACL the router switches the packet out that interface to its destination.
- If there is an ACL the router checks the packet against the access list statements sequentially. Then permits or denys each packet as it is matched.
- If the packet does not match any statement written in the ACL it is denyed because there is an implicit “deny any” statement at the end of every ACL.

Standard Access Lists

Standard Access Lists...

...are numbered from 1 to 99.

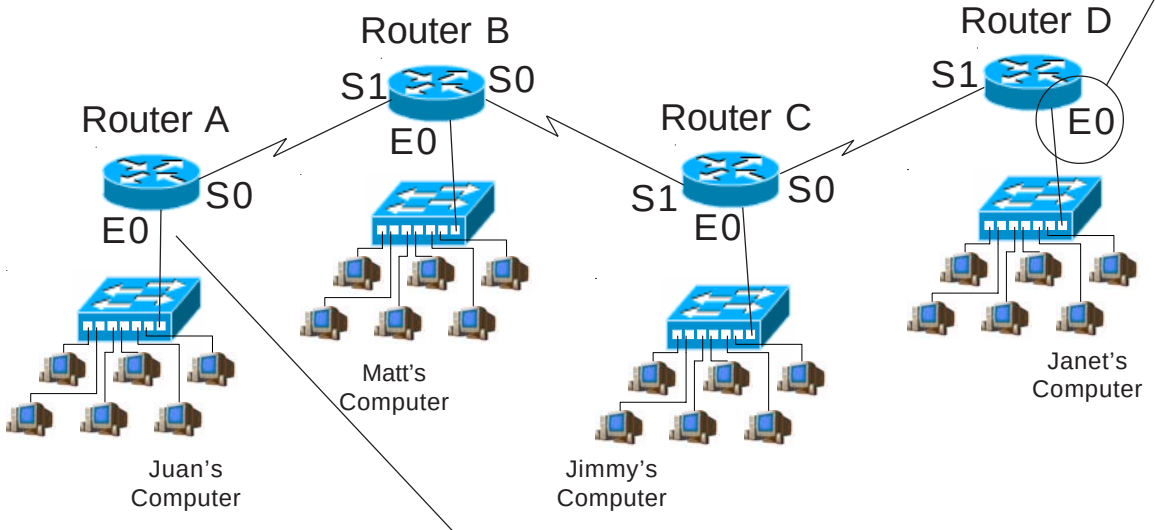
...filter (permit or deny) only source addresses.

...do not have any destination information so it must placed as close to the destination as possible.

...work at layer 3 of the OSI model.

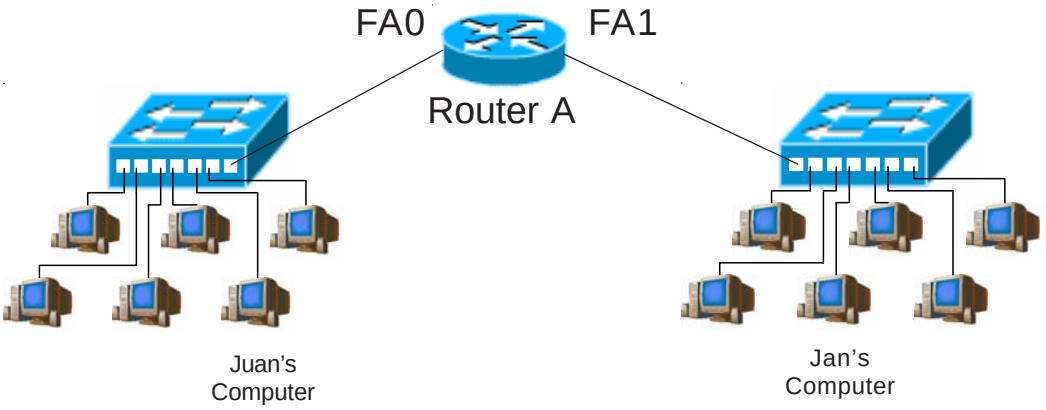
Why standard ACLs are placed close to the destination.

If you want to block traffic from Juan's computer from reaching Janet's computer with a standard access list you would place the ACL **close to the destination** on Router D, interface E0. Since its using only the source address to permit or deny packets the ACL here will not effect packets reaching Routers B, or C.

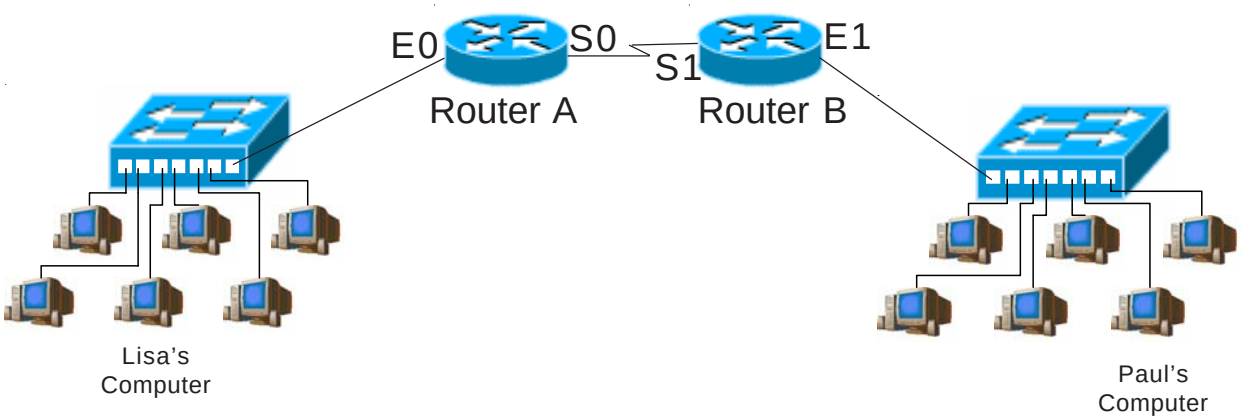


If you place the ACL on router A to block traffic to Router D it will also block all packets going to Routers B, and C; because all the packets will have the same source address.

Standard Access List Placement
Sample Problems



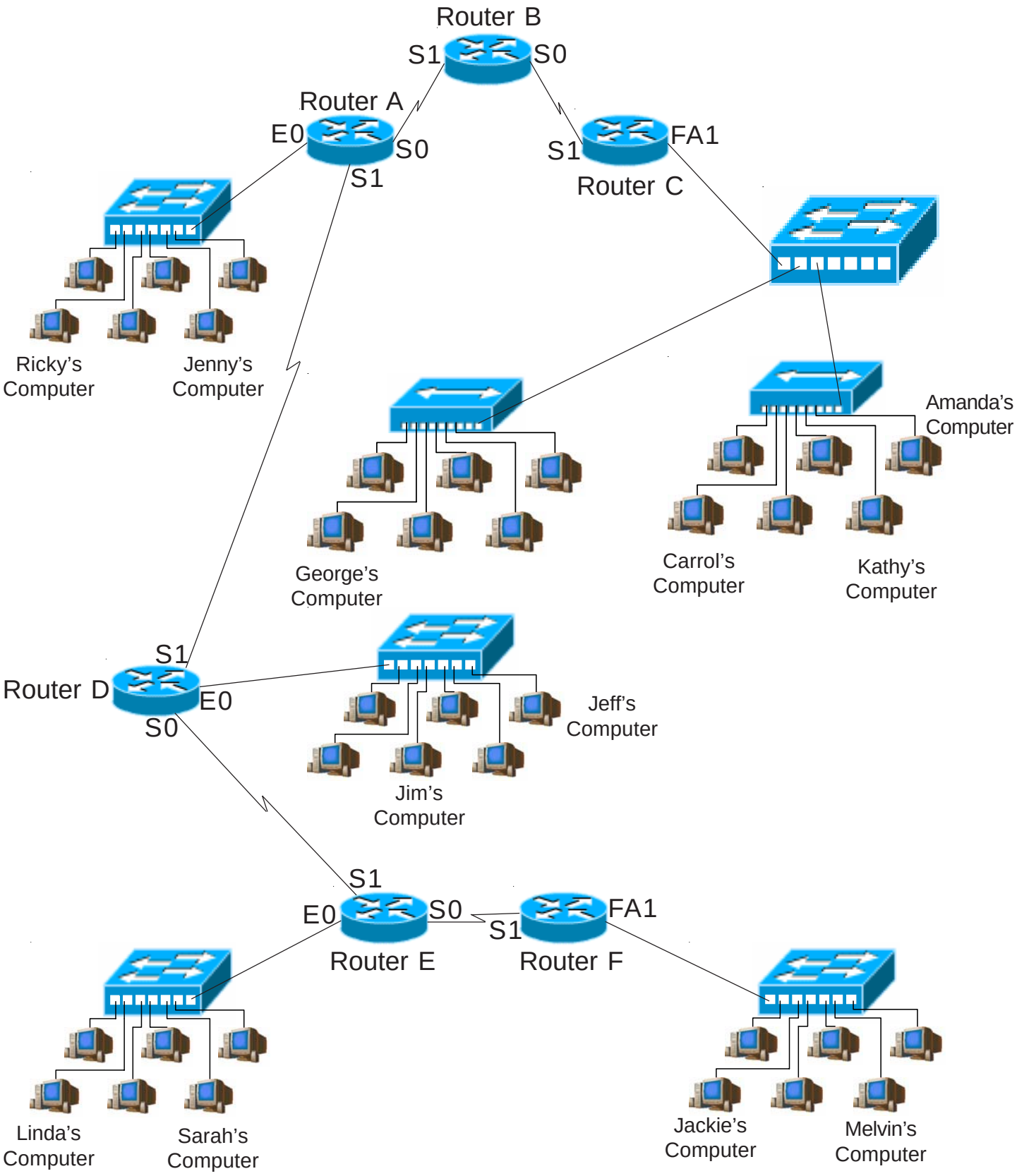
In order to permit packets from Juan's computer to arrive at Jan's computer you would place the standard access list at router interface FA1.



Lisa has been sending unnecessary information to Paul. Where would you place the standard ACL to deny all traffic from Lisa to Paul?
Router Name Router B Interface E1

Where would you place the standard ACL to deny traffic from Paul to Lisa?
Router Name Router A Interface E0

Standard Access List Placement



Standard Access List Placement

1. Where would you place a standard access list to permit traffic from Ricky's computer to reach Jeff's computer?

Router Name Router D
Interface E0

2. Where would you place a standard access list to deny traffic from Melvin's computer from reaching Jenny's computer?

Router Name Router A
Interface E0

3. Where would you place a standard access list to deny traffic to Carrol's computer from Sarah's computer?

Router Name Router C
Interface FA1

4. Where would you place a standard access list to permit traffic from Ricky's computer to reach Jeff's computer?

Router Name Router D
Interface E0

5. Where would you place a standard access list to deny traffic from Amanda's computer from reaching Jeff and Jim's computer?

Router Name Router D
Interface E0

6. Where would you place a standard access list to permit traffic from Jackie's computer to reach Linda's computer?

Router Name Router E
Interface E0

7. Where would you place a standard access list to permit traffic from Ricky's computer to reach Carrol and Amanda's computer?

Router Name Router C
Interface FA1

8. Where would you place a standard access list to deny traffic to Jenny's computer from Jackie's computer?

Router Name Router A
Interface E0

9. Where would you place a standard access list to permit traffic from George's computer to reach Linda and Sarah's computer?

Router Name Router E
Interface E0

10. Where would you place an ACL to deny traffic from Jeff's computer from reaching George's computer?

Router Name Router C
Interface FA1

11. Where would you place a standard access list to deny traffic to Sarah's computer from Ricky's computer?

Router Name Router E
Interface E0

12. Where would you place an ACL to deny traffic from Linda's computer from reaching Jackie's computer?

Router Name Router F
Interface FA1

Extended Access Lists

Extended Access Lists...

...are numbered from 100 to 199.

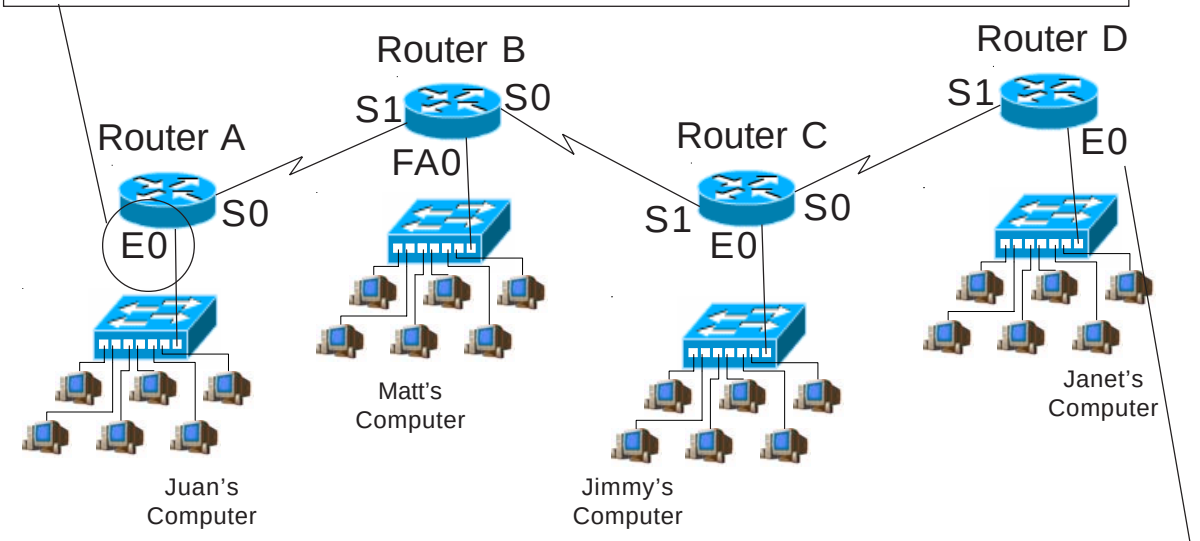
...filter (permit or deny) based on the: source address
destination address
protocol
port number

... are placed close to the source.

...work at both layer 3 and 4 of the OSI model.

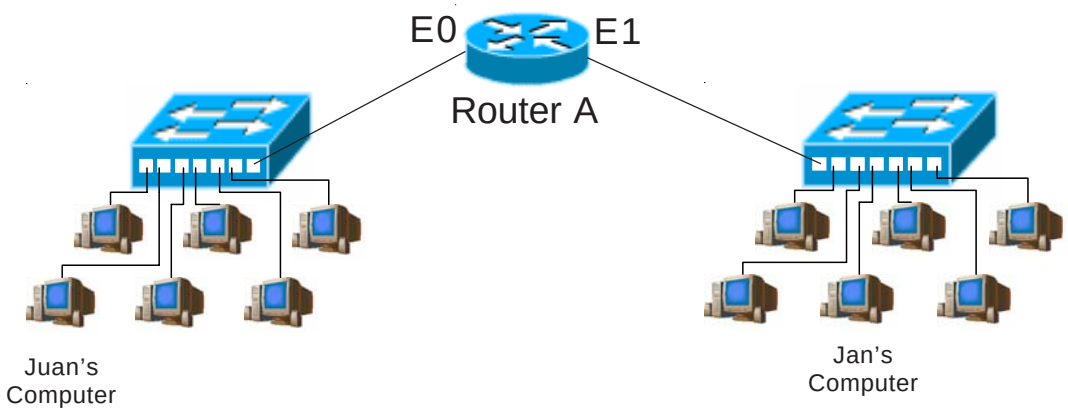
Why extended ACLs are placed close to the source.

If you want to deny traffic from Juan's computer from reaching Janet's computer with an extended access list you would place the ACL **close to the source** on Router A, interface E0. Since it can permit or deny based on the destination address it can reduce backbone overhead and not effect traffic to Routers B, or C.

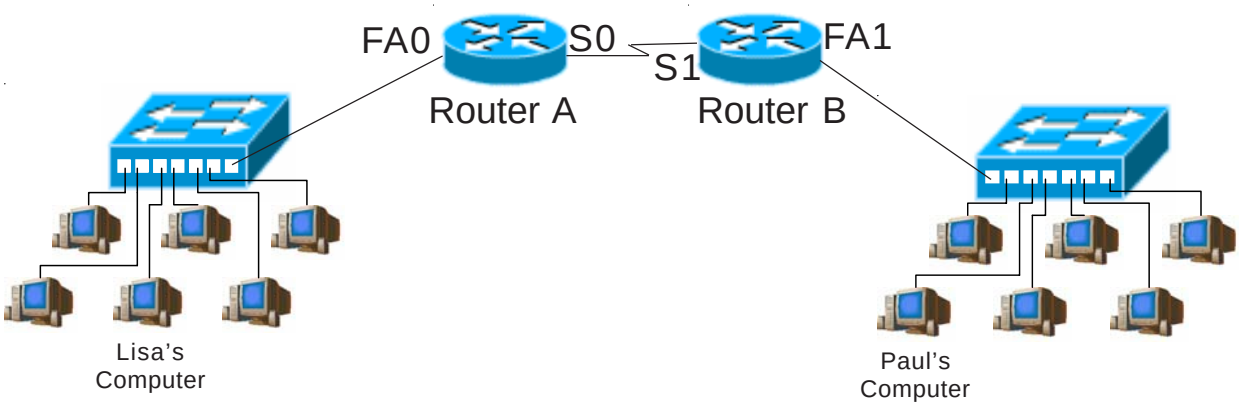


If you place the ACL on Router E to block traffic from Router A, it will work. However, Routers B, and C will have to route the packet before it is finally blocked at Router E. This increases the volume of useless network traffic.

Extended Access List Placement
Sample Problems



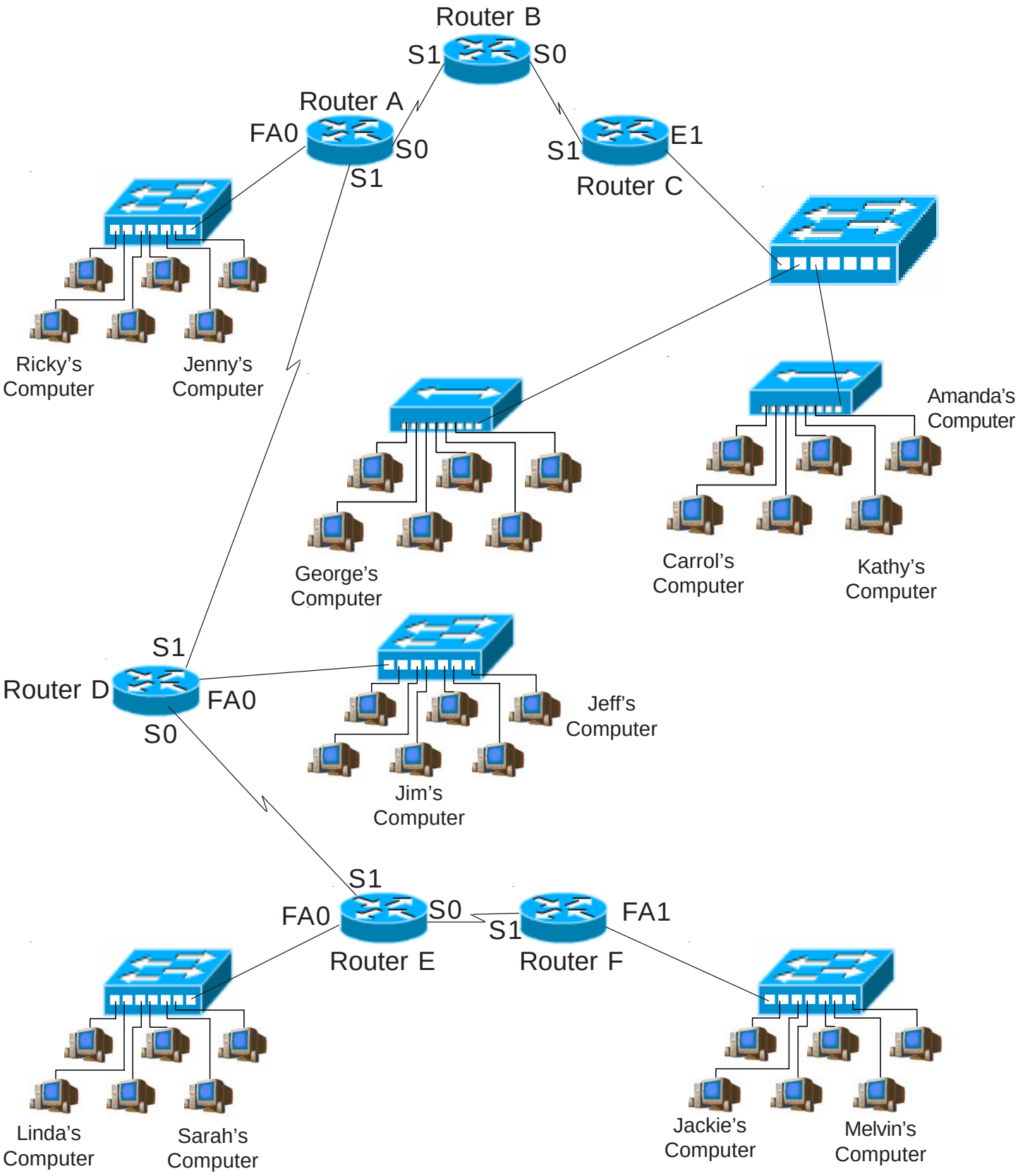
In order to permit packets from Juan's computer to arrive at Jan's computer you would place the extended access list at router interface E0.



Lisa has been sending unnecessary information to Paul. Where would you place the extended ACL to deny all traffic from Lisa to Paul?
Router Name Router A Interface FA0

Where would you place the extended ACL to deny traffic from Paul to Lisa?
Router Name Router B Interface FA1

Extended Access List Placement



Extended Access List Placement

1. Where would you place an ACL to deny traffic from Jeff's computer from reaching George's computer?
Router Name Router D
Interface Fa0

2. Where would you place an extended access list to permit traffic from Jackie's computer to reach Linda's computer?
Router Name Router F
Interface Fa1

3. Where would you place an extended access list to deny traffic to Carrol's computer from Ricky's computer?
Router Name Router A
Interface Fa0

4. Where would you place an extended access list to deny traffic to Sarah's computer from Jackie's computer?
Router Name Router F
Interface Fa1

5. Where would you place an extended access list to permit traffic from Carrol's computer to reach Jeff's computer?
Router Name Router C
Interface E1

6. Where would you place an extended access list to deny traffic from Melvin's computer from reaching Jeff and Jim's computer?
Router Name Router F
Interface Fa1

7. Where would you place an extended access list to permit traffic from George's computer to reach Jeff's computer?
Router Name Router C
Interface E1

8. Where would you place an extended access list to permit traffic from Jim's computer to reach Carrol and Amanda's computer?
Router Name Router D
Interface Fa0

9. Where would you place an ACL to deny traffic from Linda's computer from reaching Kathy's computer?
Router Name Router E
Interface Fa0

10. Where would you place an extended access list to deny traffic to Jenny's computer from Sarah's computer?
Router Name Router E
Interface Fa0

11. Where would you place an extended access list to permit traffic from George's computer to reach Linda and Sarah's computer?
Router Name Router C
Interface E1

12. Where would you place an extended access list to deny traffic from Linda's computer from reaching Jenny's computer?
Router Name Router E
Interface Fa0

Choosing to Filter Incoming or Outgoing Packets

Access Lists on your incoming port...

...requires less CPU processing.

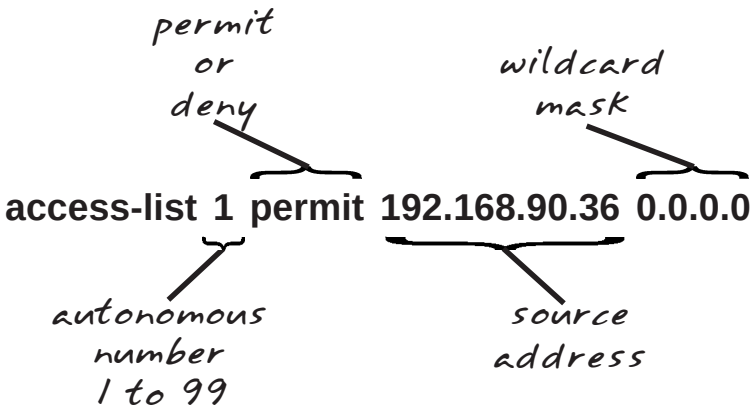
...filters and denys packets before the router has to make a routing decision.

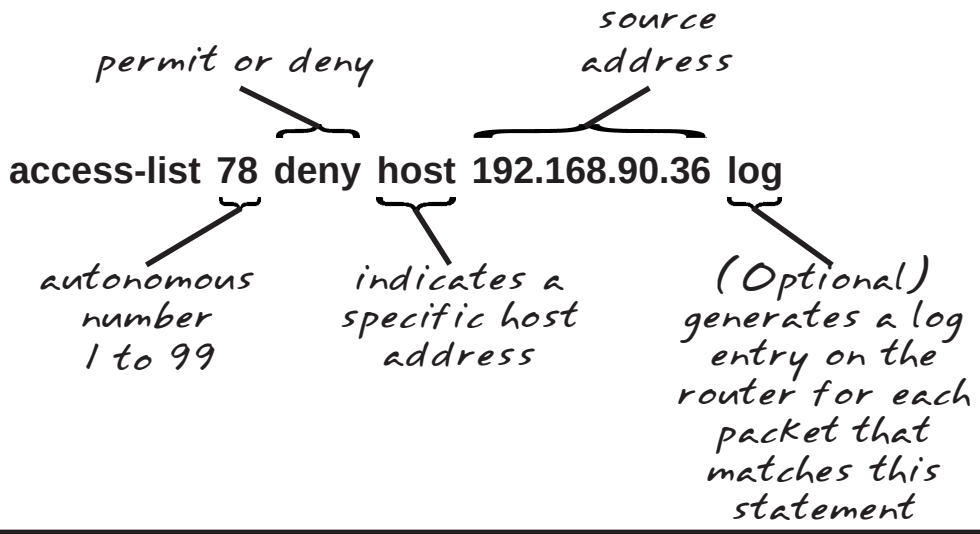
Access Lists on your outgoing port...

...are outbound by default unless otherwise specified.

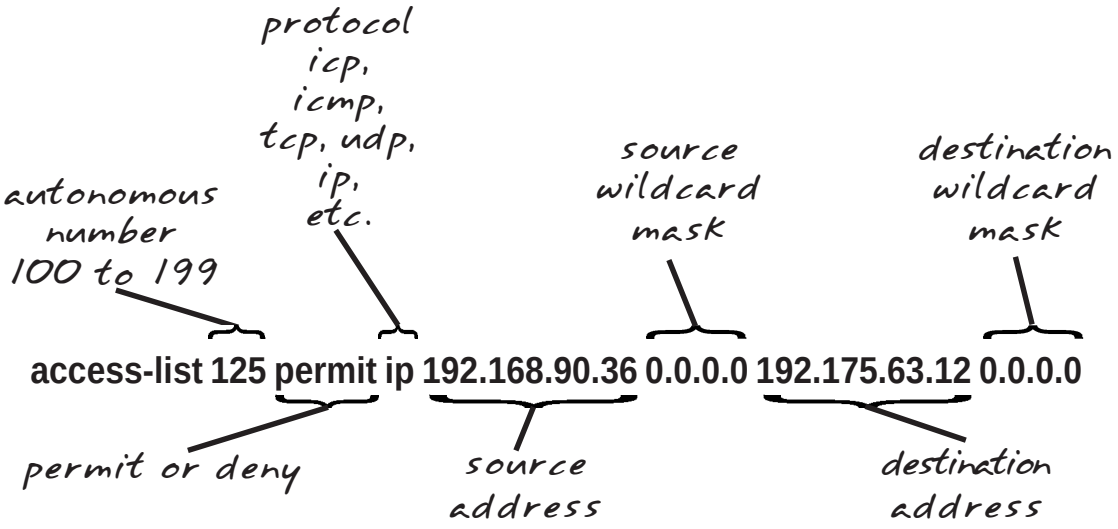
...increases the CPU processing time because the routing decision is made and the packet switched to the correct outgoing port before it is tested against the ACL.

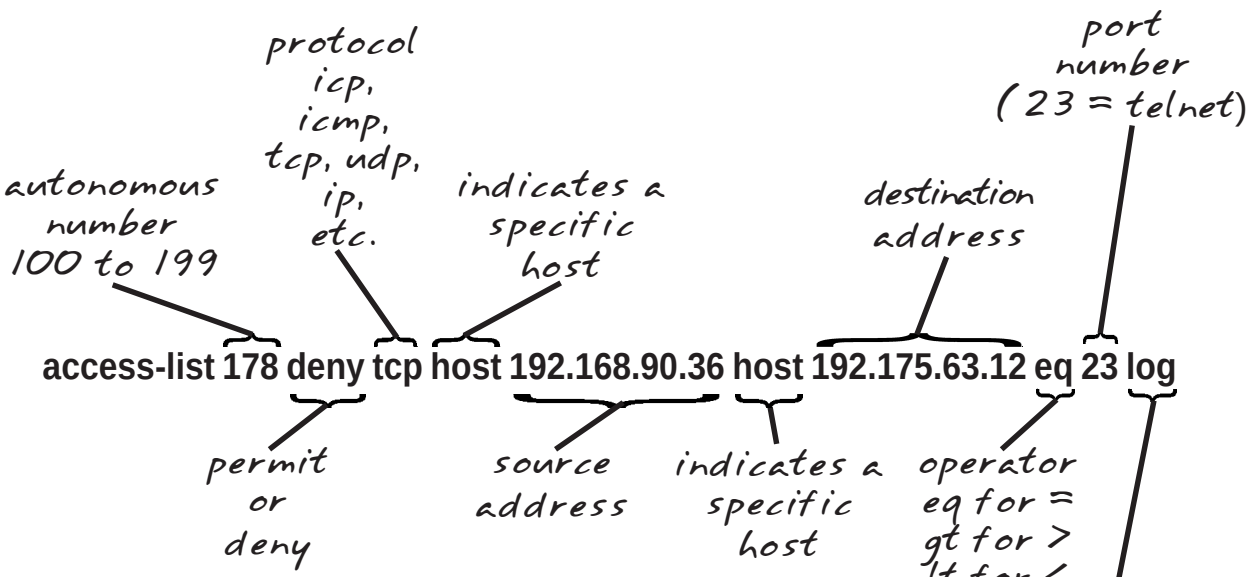
Breakdown of a Standard ACL Statement





Breakdown of an Extended ACL Statement





Protocols Include:

IP	IGMP	IPINIP
TCP	GRE	OSPF
UDP	IGRP	NOS
ICMP	EIGRP	Integer 0-255

To match any internet protocol use IP.

What are Named Access Control Lists?

Named ACLs...

...are standard or extended ACLs which have an alphanumeric name instead of a number. (ie. 1-99 or 100-199)

Named Access Lists Information

Named Access Lists...

...identify ACLs with an intuutive name instead of a number.

...eliminate the limits imposed by using numbered ACLs. (798 for standard and 799 for extended)

...provide the ability to modify your ACLs without deleting and reloading the revised access list. It will only allow you to add statements to the end of the exsisting statements.

...are not compatable with any IOS prior to Release 11.2.

...can not repeat the same name on multiple ACLs.

Applying a Standard Named Access List called "George"

Write a named standard access list called "George" on Router A, interface E1 to block Melvin's computer from sending information to Kathy's computer; but will allow all other traffic.

Place the access list at:
Router Name: Router A
Interface: E1
Access-list Name: George

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)#ip access-list standard George
Router(config-std-nacl)# deny host 72.16.70.35
Router(config-std-nacl)# access-list permit any
Router(config-std-nacl)# interface e1
Router(config-if)# ip access-group George out
Router(config-if)# exit
Router(config)# exit
```

Applying an extended Named Access List called “Gracie”

Write a named extended access list called “Gracie” on Router A, Interface E0 called “Gracie” to deny HTTP traffic intended for web server 192.168.207.27, but will permit all other HTTP traffic to reach the only the 192.168.207.0 network. Deny all other IP traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: E0
Access-list Mail: Gracie

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)#ip access-list extended Gracie
Router(config-ext-nacl)# deny tcp any host 192.168.207.27 eq www
Router(config-ext-nacl)# permit tcp any 192.168.207.0 0.0.0.255 eq www
Router(config-ext-nacl)# interface e0
Router(config-if)# ip access-group Gracie in
Router(config-if)# exit
Router(config)# exit
```

Choices for Using Wildcard Masks

Wildcard masks are usually set up to do one of four things:

1. Match a specific host.
2. Match an entire subnet.
3. Match a specific range.
4. Match all addresses.

1. Matching a specific host.

For standard access lists:

Access-List 10 permit 192.168.150.50 0.0.0.0

or

Access-List 10 permit 192.168.150.50 (standard ACL's assume a 0.0.0.0 mask)

or

Access-List 10 permit host 192.168.150.50

For extended access lists:

Access-list 110 deny ip 192.168.150.50 0.0.0.0 any

or

Access-list 110 deny ip host 192.168.150.50 any

2. Matching an entire subnet

Example 1

Address: 192.168.50.0 Subnet Mask: 255.255.255.0

Access-list 25 deny 192.168.50.0 0.0.0.255

Example 2

Address: 172.16.0.0 Subnet Mask: 255.255.0.0

Access-list 12 permit 172.16.0.0 0.0.255.255

Example 3

Address: 10.0.0.0 Subnet Mask: 255.0.0.0

Access-list 125 deny udp 10.0.0.0 0.255.255.255 any

3. Match a specific range

Example 1

Address: 10.250.50.112 Subnet Mask: 255.255.255.224

```
                                255.255.255.255
Custom Subnet mask:            -255.255.255.224
                                ---------------
Wildcard:                         0.   0.   0.  31
```

Access-list 125 permit udp 10.250.50.112 0.0.0.31 any

Example 2

Address Range: 192.168.16.0 to 192.168.16.127

```
                                192.168.16.127
                               -192.168.16.  0
                                -------------
Wildcard:                         0.   0.  0.127
```

Access-list 125 deny ip 192.168.16.0 0.0.0.127 any
(This ACL would block the lower half of the subnet.)

Example 3

Address: 172.250.16.32 to 172.250.31.63

```
                                172.250.31.  63
                               -172.250.16.  32
                                -------------
Wildcard:                         0.   0.15.  31
```

Access-list 125 permit ip 172.250.16.32 0.0.15.31 any

4. Match everyone.

For standard access lists:

Access-List 15 permit any
or
Access-List 15 deny 0.0.0.0 255.255.255.255

For extended access lists:

Access-List 175 permit ip any any
or
Access-List 175 deny tcp 0.0.0.0 255.255.255.255 any

Creating Wildcard Masks

- Just like a subnet mask the wildcard mask tells the router what part of the address to check or ignore. Zero (0) must match exactly, one (1) will be ignored.
- The source address can be a single address, a range of addresses, or an entire subnet.
- As a rule of thumb the wildcard mask is the reverse of the subnet mask.

Example #1:
IP Address and subnet mask: 204.100.100.0 255.255.255.0
IP Address and wildcard mask: 204.100.100.0 0.0.0.255

- All zero's (or 0.0.0.0) means the address must match exactly.

Example #2:
10.10.150.95 0.0.0.0 (This address must match exactly.)

- One's will be ignored.

Example #3:
10.10.150.95 0.0.0.255 (Any 10.10.150.0 subnet address will match.
10.10.150.0 to 10.10.150.255)

- This also works with subnets.

Example #4:
IP Address and subnet mask: 192.170.25.30 255.255.255.224
IP Address and wildcard mask: 192.170.25.30 0.0.0.31
(Subtract the subnet mask from
255.255.255.255 to create the wildcard)

Do the math... 255 - 255 = 0 (This is the inverse of the subnet mask.)
255 - 224 = 31

Example #5:
IP Address and subnet mask: 172.24.128.0 255.255.128.0
IP Address and wildcard mask: 172.24.128.0 0.0.127.255

Do the math... 255 - 255 = 0 (This is the inverse of the subnet mask.)
255 - 128 = 127
255 - 0 = 255

Wildcard Mask Problems

1. Create a wildcard mask to match this exact address.
   IP Address: 192.168.25.70
   Subnet Mask: 255.255.255.0
   0.0.0.0

2. Create a wildcard mask to match this range.
   IP Address: 210.150.10.0
   Subnet Mask: 255.255.255.0
   0.0.0.255

3. Create a wildcard mask to match this host.
   IP Address: 195.190.10.35
   Subnet Mask: 255.255.255.0
   0.0.0.0

4. Create a wildcard mask to match this range.
   IP Address: 172.16.0.0
   Subnet Mask: 255.255.0.0
   0.0.255.255

5. Create a wildcard mask to match this range.
   IP Address: 10.0.0.0
   Subnet Mask: 255.0.0.0
   0.255.255.255

6. Create a wildcard mask to match this exact address.
   IP Address: 165.100.0.130
   Subnet Mask: 255.255.255.192
   0.0.0.0

7. Create a wildcard mask to match this range.
   IP Address: 192.10.10.16
   Subnet Mask: 255.255.255.224
   0.0.0.31

8. Create a wildcard mask to match this range.
   IP Address: 171.50.75.128
   Subnet Mask: 255.255.255.192
   0.0.0.63

9. Create a wildcard mask to match this host.
   IP Address: 10.250.30.2
   Subnet Mask: 255.0.0.0
   0.0.0.0

10. Create a wildcard mask to match this range.
    IP Address: 210.150.28.16
    Subnet Mask: 255.255.255.248
    0.0.0.7

11. Create a wildcard mask to match this range.
    IP Address: 172.18.0.0
    Subnet Mask: 255.255.224.0
    0.0.31.255

12. Create a wildcard mask to match this range.
    IP Address: 135.35.230.32
    Subnet Mask: 255.255.255.248
    0.0.0.7

Wildcard Mask Problems

Based on the given information list the usable source addresses or range of usable source addresses that would be permitted or denied for each access list statement.

1. access-list **10 permit 192.168.150.50 0.0.0.0**

Answer: 192.168.150.50

2. **access-list 5 permit any**

Answer: Any address

3. **access-list 125 deny tcp 195.223.50.0 0.0.0.63 host 172.168.10.1 fragments**

Answer: 195.223.50.1 to 195.223.50.63

4. **access-list 11 deny 210.10.10.0 0.0.0.255**

Answer: 210.10.10.1 to 210.10.10.254

5. **access-list 108 deny ip 192.220.10.0 0.0.0.15 172.32.4.0 0.0.0.255**

Answer: 192.220.10.1 to 192.220.10.15

6. **access-list 171 deny any host 175.18.24.10 fragments**

Answer: Any Address

7. **access-list 105 permit 192.168.15.0 0.0.0.255 any**

Answer: 192.168.15.1 to 192.168.15.254

8. **access-list 109 permit tcp 172.16.10.0 0.0.0.255 host 192.168.10.1 eq 80**

Answer: 172.16.10.1 to 172.16.10.254

9. **access-list 111 permit ip any any**

Answer: Any Address

10. **access-list 195 permit udp 172.30.12.0 0.0.0.127 172.50.10.0 0.0.0.255**

Answer: 172.30.12.1 to 172.30.12.127

11. **access-list 110 permit ip 192.168.15.0 0.0.0.3 192.168.30.10 0.0.0.0**

Answer: 192.168.15.1 to 192.168.15.3

12. **access-list 120 permit ip 192.168.15.0 0.0.0.7 192.168.30.10 0.0.0.0**

Answer: 192.168.15.1 to 192.168.15.7

13. **access-list 130 permit ip 192.168.15.0 0.0.0.15 192.168.30.10 0.0.0.0**

Answer: 192.168.15.1 to 192.168.15.15

14. **access-list 140 permit ip 192.168.15.0 0.0.0.31 192.168.30.10 0.0.0.0**

Answer: 192.168.15.1 to 192.168.15.31

15. **access-list 150 permit ip 192.168.15.0 0.0.0.63 192.168.30.10 0.0.0.0**

Answer: 192.168.15.1 to 192.168.15.63

16. **access-list 101 Permit ip 192.168.15.0 0.0.0.127 192.168.30.10 0.0.0.0**

Answer: 192.168.15.1 to 192.168.15.127

17. **access-list 185 permit ip 192.168.15.0 0.0.0.255 192.168.30.0 0.0.0.255**

Answer: 192.168.15.1 to 192.168.15.254

18. **access-list 160 deny udp 172.16.0.0 0.0.1.255 172.18.10.18 0.0.0.0 gt 22**

Answer: 172.16.0.1 to 172.16.1.254

19. **access-list 195 permit icmp 172.85.0.0 0.0.15.255 172.50.10.0 0.0.0.255**

Answer: 172.85.0.1 to 172.85.15.254

20. **access-list 10 permit 175.15.120.0 0.0.0.255**

Answer: 175.15.120.1 to 175.15.120.254

21. **access-list 190 permit tcp 172.15.0.0 0.0.15.31 any**

Answer: 172.15.0.1 to 172.15.15.31

22. **access-list 100 permit ip 10.0.0.0 0.255.255.255 172.50.10.0 0.0.0.255**

Answer: 10.0.0.1 to 10.255.255.254

Wildcard Mask Problems

Based on the given information list the usable destination addresses or range of usable destination addresses that would be permitted or denied for each access list statement.

1.**access-list 125 deny tcp 195.223.50.0 0.0.0.63 host 172.168.10.1 fragments**

Answer: 172.168.10.1

2. **access-list 115 permit any any**

Answer: Any address

3. **access-list 150 permit ip 192.168.30.10 0.0.0.0 192.168.15.0 0.0.0.63**

Answer: 192.168.15.1 to 192.168.15.63

4. **access-list 120 deny tcp 172.32.4.0 0.0.0.255 192.220.10.0 0.0.0.15**

Answer: 192.220.10.1 to 192.220.10.15

5. **access-list 108 deny ip 192.220.10.0 0.0.0.15 172.32.4.0 0.0.0.255**

Answer: 172.32.4.1 to 172.32.4.254

6. **access-list 101 deny ip 140.130.110.100 0.0.0.0 0.0.0.0 255.255.255.255**

Answer: Any Address

7. **access-list 105 permit any 192.168.15.0 0.0.0.255**

Answer: 192.168.15.1 to 192.168.15.254

8. **access-list 120 permit ip 192.168.15.10 0.0.0.0 192.168.30.0 0.0.0.7**

Answer: 192.168.30.1 to 192.168.30.7

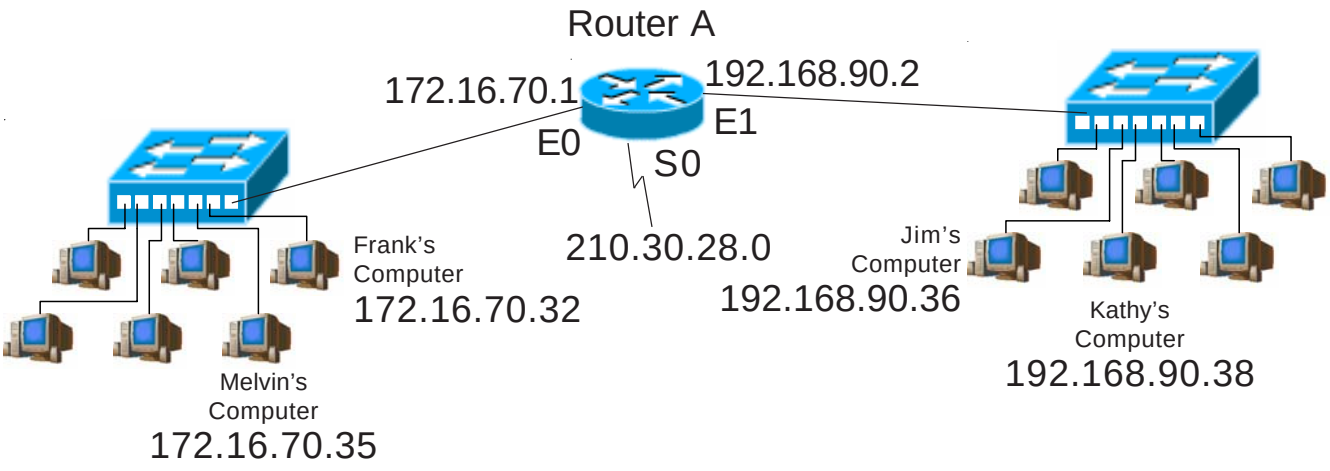
9. **access-list 160 deny udp 172.16.0.0 0.0.1.255 172.18.10.18 0.0.0.0 eq 21**

Answer: 172.18.10.18

10. **access-list 150 permit ip 192.168.15.10 0.0.0.0 192.168.30.0 0.0.0.63**

Answer: 192.168.30.1 to 192.168.30.63

Writing
Standard Access Lists...



Standard Access List Sample #1

Write a standard access list to block Melvin's computer from sending information to Kathy's computer; but will allow all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: E1
Access-list #: 10

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)# access-list 10 deny 172.16.70.35
                                    or
                access-list 10 deny 172.16.70.35 0.0.0.0
                                    or
                access-list 10 deny host 172.16.70.35
Router(config)# access-list 10 permit 0.0.0.0 255.255.255.255
                                    or
                access-list 10 permit any
Router(config)# interface e1
Router(config-if)# ip access-group 10 out
Router(config-if)# exit
Router(config)# exit
```

[Viewing information about existing ACL's]

`Router#` show configuration	(This will show which access groups are associated with particular interfaces)
`Router#` show access list 10	(This will show detailed information about this ACL)

Standard Access List Sample #2

Write a standard access list to block Jim's computer from sending information to Frank's computer; but will allow all other traffic from the 192.168.90.0 network. Permit all traffic from the 210.30.28.0 network to reach the 172.16.70.0 network. Deny all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: E0
Access-list #: 28

[Writing and installing an ACL]

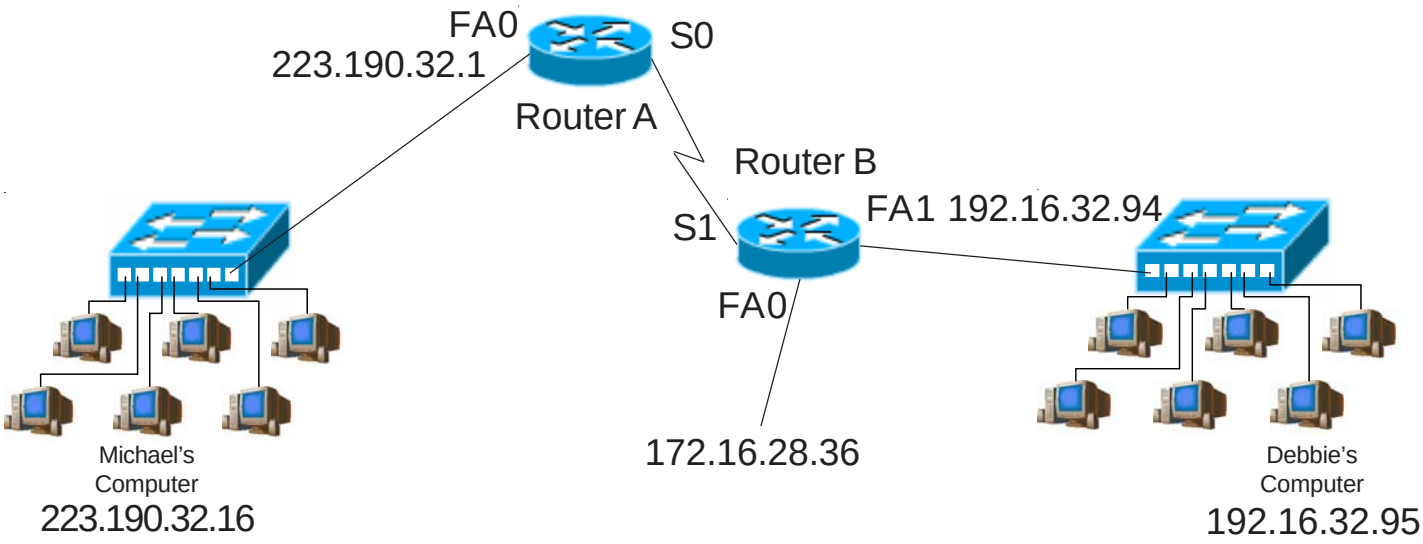
```
Router# configure terminal
Router(config)# access-list 28 deny 192.168.90.36
                                    or
                access-list 28 deny 192.168.90.36 0.0.0.0
                                    or
                access-list 28 deny host 192.168.90.36
Router(config)# access-list 28 permit 192.168.90.0 0.0.0.255
Router(config)# access-list 28 permit 210.30.28.0 0.0.0.255
Router(config)# interface e0
Router(config-if)# ip access-group 28 out
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```

[Disabling ACL's]

```
Router# configure terminal
Router(config)# interface e0
Router(config-if)# no ip access-group 28 out
Router(config-if)# exit
Router(config)# exit
```

[Removing an ACL]

```
Router# configure terminal
Router(config)# interface e0
Router(config-if)# no ip access-group 28 out
Router(config-if)# exit
Router(config)# no access-list 28
Router(config)# exit
```



Standard Access List Problem #1

Write a standard access list to block Debbie's computer from receiving information from Michael's computer; but will allow all other traffic. List all the command line options for this problem. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: FA1
Access-list #: 35 (1-99)

[Writing and installing an ACL]

```
Router# configure terminal (or config t)

Router(config)# access-list 35 deny 223.190.32.16
                or
                access-list 35 deny host 223.190.32.16
                or
                access-list 35 deny 223.190.32.16 0.0.0.0

Router(config)# access-list 35 permit any
                or
                access-list 35 permit 0.0.0.0 255.255.255.255

Router(config)# interface FA1

Router(config-if)# ip access-group 35 in or (out) (circle one)
Router(config-if)# exit
Router(config)# exit
```

Standard Access List Problem #2

Write a standard access list to permit Debbie's computer to receive information from Michael's computer; but will deny all other traffic from the 224.190.32.0 network. Block all traffic from the 172.16.0.0 network. Permit all other traffic. List all the command line options for this problem. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: FA0
Access-list #: 40 (1-99)

[Writing and installing an ACL]

Router# configure terminal (or config t)

Router(config)# access-list 40 permit 223.190.32.16
or
access-list 40 permit host 223.190.32.16
or
access-list 40 permit 223.190.32.16 0.0.0.0

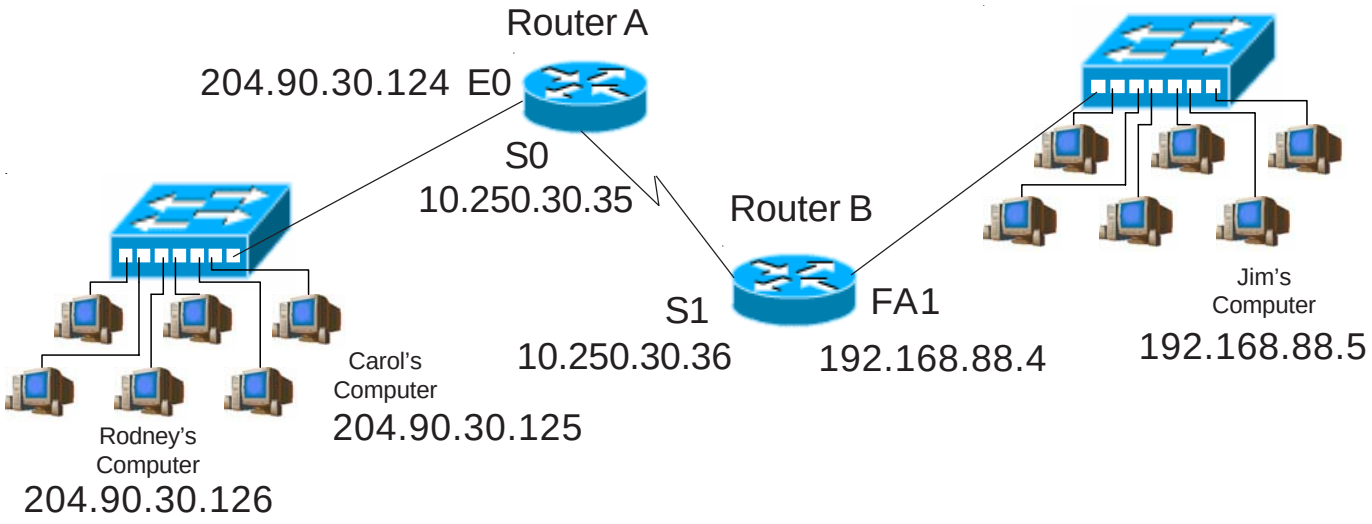
Router(config)# access-list 40 deny 223.190.32.0 0.0.0.255

Router(config)# access-list 40 deny 172.16.0.0 0.0.255.255

Router(config)# access-list 40 permit any
or
access-list 40 permit 0.0.0.0 255.255.255.255

Router(config)# interface FA0

Router(config-if)# ip access-group 40 in or (out) (circle one)
Router(config-if)# exit
Router(config)# exit



Standard Access List Problem #3

Write a standard access list to block Rodney and Carol's computer from sending information to Jim's computer; but will allow all other traffic from the 204.90.30.0 network. Block all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: FA1
Access-list #: 45 (1-99)

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)#   access-list 45 deny 204.90.30.125
                  or
                  access-list 45 deny host 204.90.30.125
                  or
                  access-list 45 deny 204.90.30.125 0.0.0.0
                  ----------------------------------------
                  access-list 45 deny 204.90.30.126
                  or
                  access-list 45 deny host 204.90.30.126
                  or
                  access-list 45 deny 204.90.30.126 0.0.0.0
                  ----------------------------------------
                  access-list 45 permit 204.90.30.0 0.0.0.255
                  ----------------------------------------
Router(config)# interface FA1

Router(config-if)# ip access-group 45 in or (out) (circle one)
Router(config-if)# exit
Router(config)# exit
```

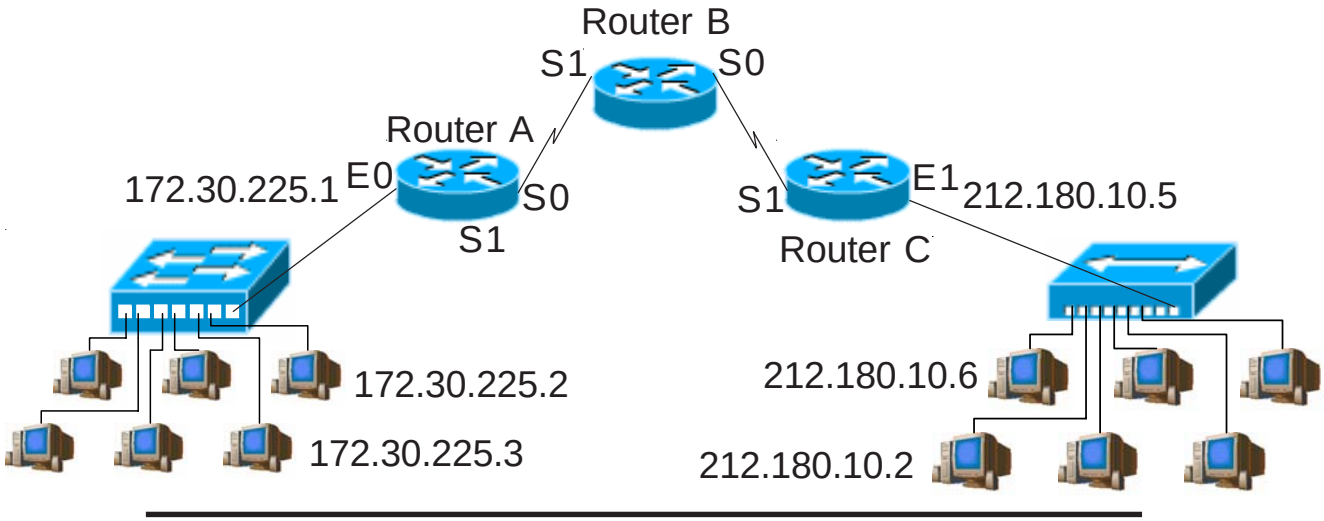
Standard Access List Problem #4

Using a minimum number of commands write a standard access list named "Ralph" to block Carol's computer from sending information to Jim's computer; but will permit Jim to receive data from Rodney. Block the upper half of the 204.90.30.0 range from reaching Jim's computer while permitting the lower half of the range. Block all other traffic. For help with blocking the upper half of the range review page 13 or the wildcard mask problems on pages 16 and 17. For help with named ACLs review pages 12 and 13.

Place the access list at:
Router Name: Router B
Interface: FA1
Access-list Name: Ralph

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)# ip access-list standard Ralph
Router(config-std-nacl)# permit 204.90.30.0 0.0.0.127
Router(config-std-nacl)# interface FA1
Router(config-if)# ip access-group Ralph in or (out) (circle one)
Router(config-if)# exit
Router(config)# exit
```



Standard Access List Problem #5

Write a standard access list to block 172.30.225.2 and 172.30.225.3 from sending information to the 212.180.10.0 network; but will allow all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router C
Interface: E1
Access-list #: 55 (1-99)

[Writing and installing an ACL]

```
Router# configure terminal (or config t)

Router(config)#   access-list 55 deny 172.30.225.2
                  or
                  access-list 55 deny host 172.30.225.2
                  or
                  access-list 55 deny 172.30.225.2 0.0.0.0

                  access-list 55 deny 172.30.225.3
                  or
                  access-list 55 deny host 172.30.225.3
                  or
                  access-list 55 deny 172.30.225.3 0.0.0.0

                  access-list 55 permit any

Router(config)# interface E1

Router(config-if)# ip access-group 55 in or (out) (circle one)
Router(config-if)# exit
Router(config)# exit
```

Standard Access List Problem #6

Write a standard access list to block and log 212.180.10.2 from sending information to the 172.30.225.0 network. Permit and log 212.180.10.6 to send data to the 172.30.225.0 network. Deny all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written. (Check the example on page 10 for help with the logging option.)

Place the access list at:
Router Name: Router A
Interface: E0
Access-list #: 60 (1-99)

[Writing and installing an ACL]

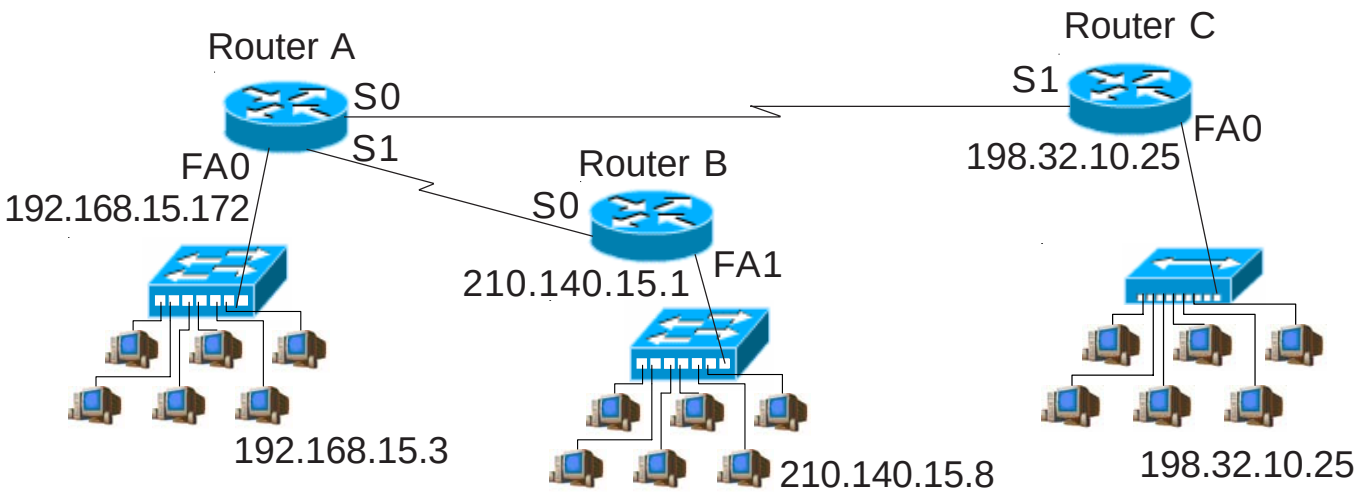
```
Router# configure terminal (or config t)

Router(config)#  access-list 60 deny 212.180.10.2 log
                 or
                 access-list 60 deny host 212.180.10.2 log
                 or
                 access-list 60 deny 212.180.10.2 0.0.0.0 log

                 access-list 60 permit 212.180.10.6 log
                 or
                 access-list 60 permit host 212.180.10.6 log
                 or
                 access-list 60 permit 212.180.10.6 0.0.0.0 log

Router(config)# interface E0

Router(config-if)# ip access-group 60 in or out (circle one)
Router(config-if)# exit
Router(config)# exit
```



Standard Access List Problem #7

Write a standard access list to block the addresses 192.168.15.1 to 192.168.15.31 from sending information to the 210.140.15.0 network. Do not permit any traffic from 198.32.10.25 to reach the 210.140.15.0 network. Permit all other traffic. For help with this problem review page 13 or the wildcard mask problems on pages 16 and 17.

Place the access list at:
Router Name: Router B
Interface: FA1
Access-list #: 65 (1-99)

[Writing and installing an ACL]

Router# configure terminal (or config t)

Router(config)# access-list 65 deny 192.168.15.0 0.0.0.31

access-list 65 deny 198.32.10.25
or
access-list 65 deny host 198.32.10.25
or
access-list 65 deny 198.32.10.25 0.0.0.0

access-list 65 permit any

Router(config)# interface FA1

Router(config-if)# ip access-group 65 in or (out) (circle one)
Router(config-if)# exit
Router(config)# exit

Standard Access List Problem #8

Write a standard named access list called "Cisco_Lab_A" to permit traffic from the lower half of the 198.32.10.0 network to reach 192.168.15.0 network; block the upper half of the addresses. Allow host 198.32.10.192 to reach network 192.168.15.0. Permit all other traffic. For help with this problem review page 13 or the wildcard masks problems on pages 16 and 17. For assistance with named ACLs review pages 12 and 13.

Place the access list at:
Router Name: Router A
Interface: FA0
Access-list Name: Cisco_Lab_A

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)# access-list standard Cisco_Lab_A
Router(config-std-nacl)# permit 198.32.10.0 0.0.0.127
                         deny 198.32.10.0 0.0.0.255
                         permit any
Router(config-std-nacl)# interface FA0
Router(config-if)# ip access-group Cisco_Lab_A in or out (circle one)
Router(config-if)# exit
Router(config)# exit
```

Standard Access List Problem #9

Write a standard access list to block network 192.168.255.0 from receiving information from the following addresses: 10.250.1.1, 10.250.2.1, 10.250.4.1, and the entire 10.250.3.0 255.255.255.0 network. Allow all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: FA0
Access-list #: 75 (1-99)

[Writing and installing an ACL]

Router# configure terminal (or config t)

Router(config)#
access-list 75 deny 10.250.1.1
or
access-list 75 deny host 10.250.1.1
or
access-list 75 deny 10.250.1.1 0.0.0.0

access-list 75 deny 10.250.2.1
or
access-list 75 deny host 10.250.2.1
or
access-list 75 deny 10.250.2.1 0.0.0.0

access-list 75 deny 10.250.4.1
or
access-list 75 deny host 10.250.4.1
or
access-list 75 deny 10.250.4.1 0.0.0.0

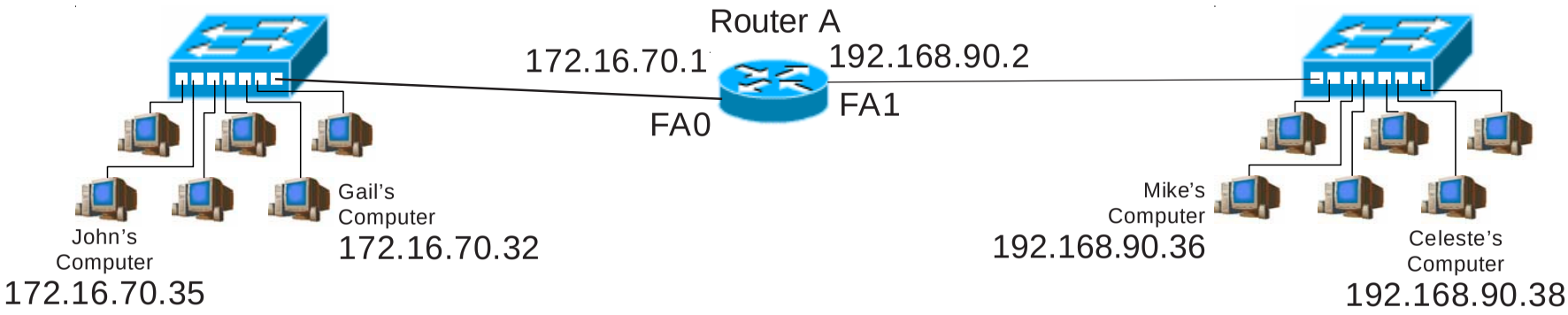
access-list 75 deny 10.250.3.0 0.0.0.255

access-list 75 permit any

Router(config)# interface FA0

Router(config-if)# ip access-group 75 in or (out) (circle one)
Router(config-if)# exit
Router(config)# exit

Writing Extended Access Lists...



Extended Access List Sample #1 — Deny/Permit Specific Addresses

Write an extended access list to prevent John's computer from sending information to Mike's computer; but will allow all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: FA0
Access-list #: 110

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)# access-list 110 deny ip 172.16.70.35 0.0.0.0 192.168.90.36 0.0.0.0
                                    or
                access-list 110 deny ip host 172.16.70.35 host 192.168.90.36
Router(config)# access-list 110 permit ip any any
                                    or
                access-list 110 permit ip 0.0.0.0 255.255.255.2550.0.0.0 255.255.255.255
Router(config)# interface fa0
Router(config-if)# ip access-group 110 in
Router(config-if)# exit
Router(config)# exit
```

[Viewing information about existing ACL's]

`Router# show configuration`	(This will show which access groups are associated with particular interfaces)
`Router# show access list 110`	(This will show detailed information about this ACL)

Extended Access List Sample #2 — Deny/Permit Specific Addresses

Write an extended access list to block the 172.16.70.0 network from receiving information from Mike's computer at 192.168.90.36. Block the lower half of the ip addresses from 192.168.90.0 network from reaching Gail's computer at 172.16.70.32. Permit all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: FA1
Access-list #: 135

[Writing and installing an ACL]

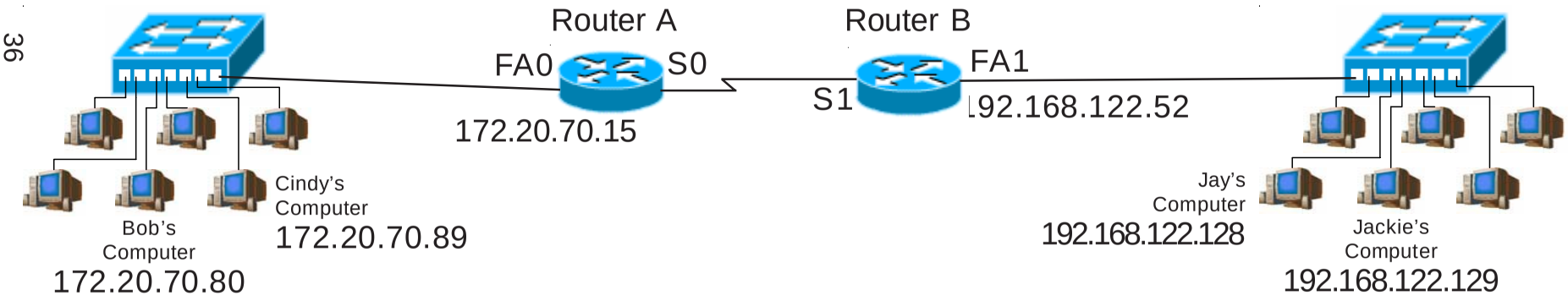
```
Router# configure terminal
Router(config)# access-list 135 deny ip 192.168.90.36 0.0.0.0 172.16.70.0 0.0.0.255
                                  or
                access-list 135 deny ip host 192.168.90.36 172.16.70.0 0.0.0.255
Router(config)# access-list 135 deny ip 192.168.90.0 0.0.0.127 172.16.70.32 0.0.0.0
                                  or
                access-list 135 deny ip 192.168.90.0 0.0.0.127 host 172.16.70.32
Router(config)# access-list 135 permit ip any any
                                  or
               access-list 135 permit ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
Router(config)# interface fa1
Router(config-if)# ip access-group 135 in
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```

[Disabling ACL's]

```
Router# configure terminal
Router(config)# interface e1
Router(config-if)# no ip access-group 135 out
Router(config-if)# exit
Router(config)# exit
```

[Removing an ACL]

```
Router# configure terminal
Router(config)# interface e1
Router(config-if)# no ip access-group 135 out
Router(config-if)# exit
Router(config)# no access-list 135
Router(config)# exit
```



Extended Access List Problem #1 — Deny/Permit Specific Addresses

Write an extended access list to prevent Jay's computer from receiving information from Cindy's computer. Permit all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: FA0
Access-list #: 105 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal (or config t)

Router(config)# access-list 105 deny ip host 172.20.70.89 host 192.168.122.128
          or
             access-list 105 deny ip 172.30.225.2 0.0.0.0 192.168.122.128 0.0.0.0

             access-list 105 permit ip any any

Router(config)# interface FA0
Router(config-if)# ip access-group 105 (in) or out (circle one)
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```

Extended Access List Problem #2 — Deny/Permit Specific Addresses

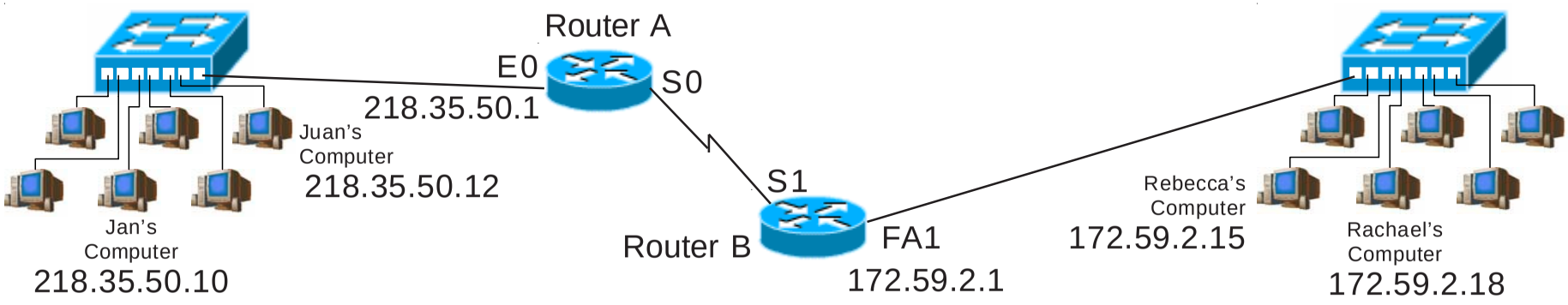
Write an extended access list to block the 172.20.70.0 255.255.255.0 network from receiving information from Jackie's computer at 192.168.122.129. Block the lower half of the ip addresses from 192.168.122.0 network from reaching Cindy's computer at 172.20.70.89. Permit all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: FA1
Access-list #: 110 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal
Router(config)# access-list 110 deny ip host 192.168.122.129 172.20.70.0 0.0.0.255
```
or
```
access-list 110 deny ip 192.168.122.129 0.0.0.0 172.20.70.0 0.0.0.255
```
```
access-list 110 deny ip 192.168.122.0 0.0.0.127 host 172.20.70.89
```
or
```
access-list 110 deny ip 192.168.122.0 0.0.0.127 172.20.70.89 0.0.0.0
access-list 110 permit ip any any
```

```
Router(config)# interface E1
Router(config-if)# ip access-group 105 (in) or out (circle one)
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```



Extended Access List Problem #3 — Deny/Permit Specific Addresses

Write a named extended access list called "Lab_166" to permit Jan's computer at 218.35.50.10 to receive packets from Rachael's computer at 172.59.2.18; but not Rebecca's computer at 172.59.2.15. Deny all other packets. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: FA1
Access-list Name: Lab_166

[Writing and installing an ACL]

```
Router# configure terminal (or config t)

Router(config)#access-list extended Lab_166

Router(config-ext-nacl)# permit ip host 172.59.2.18 host 218.35.50.10
                      or
                        permit ip 172.59.2.18 0.0.0.0 218.35.50.10 0.0.0.0


Router(config-ext-nacl)# interface FA1
Router(config-if)# ip access-group Lab_166 (in) or out (circle one)
Router(config-if)# exit
Router(config)# exit
```

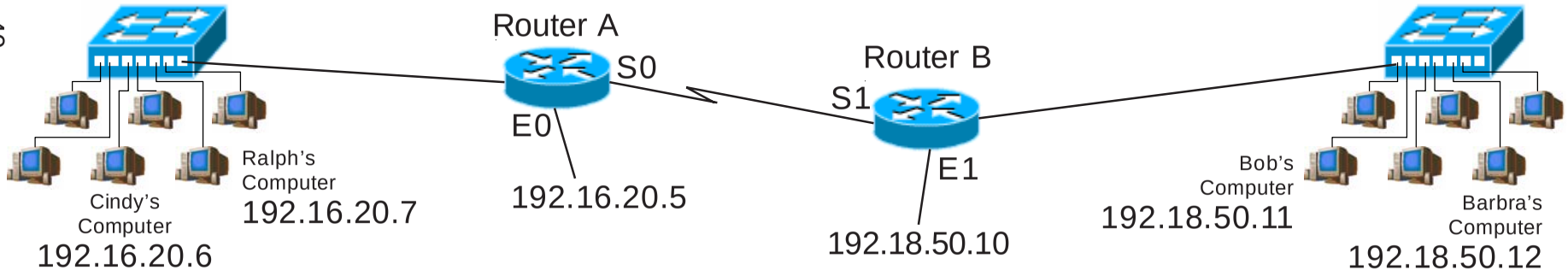
Extended Access List Problem #4 — Deny/Permit Specific Addresses

Write an extended access list to allow Juan's computer at 218.35.50.12 to send information to Rebecca's computer at 172.59.2.15; but not Rachael's computer at 172.59.2.18. Permit all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: E0
Access-list #: 120 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal
Router(config)# access-list 120 deny ip host 218.35.50.12 host 172.59.2.18
        or
          access-list 120 deny ip 218.35.50.12 0.0.0.0 172.59.2.18 0.0.0.0
          access-list 120 permit ip any any
Router(config)# interface Fa1
Router(config-if)# ip access-group 115 (in) or out (circle one)
Router((config-if)# exit
Router(config)# exit
Router# copy run start
```



Extended Access List Sample #3 — Deny/Permit Entire Ranges

Write an extended access list to permit the 192.16.20.0 network to receive packets from the 192.18.50.0 network. Deny all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: E1
Access-list #: 111

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)# access-list 111 permit ip 192.18.50.0 0.0.0.255 192.168.20.0 0.0.0.255
Router(config)# access-list 111 deny ip any any
                                  or
                access-list 111 deny ip 0.0.0.0 255.255.255.2550.0.0.0 255.255.255.255
Router(config)# interface e1
Router(config-if)# ip access-group 111 in
Router(config-if)# exit
Router(config)# exit
```

[Viewing information about existing ACL's]

`Router# show configuration`	(This will show which access groups are associated with particular interfaces)
`Router# show access list 111`	(This will show detailed information about this ACL)

Extended Access List Sample #4 — Deny/Permit Entire Ranges

Write an extended access list to block the 192.18.50.0 network from receiving information from the 192.16.20.0 network. Permit all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: E0
Access-list #: 188

[Writing and installing an ACL]

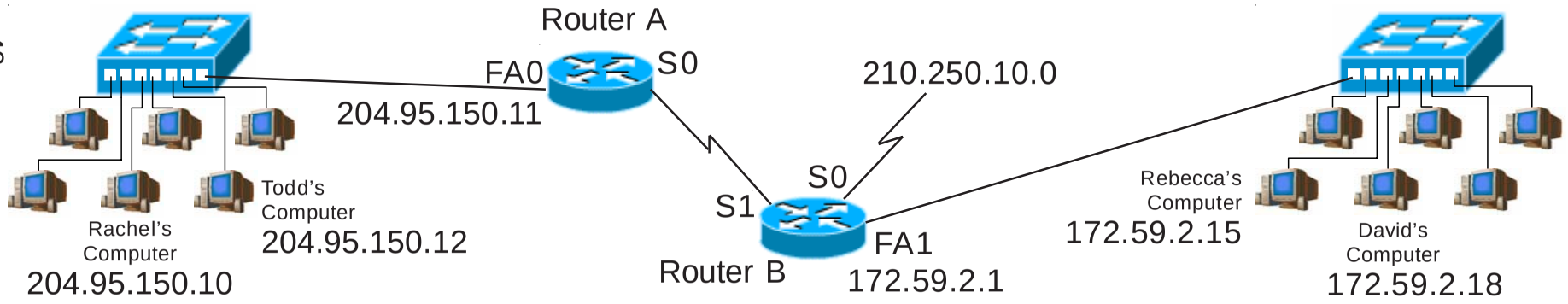
```
Router# configure terminal
Router(config)# access-list 188 deny ip 192.16.20.0 0.0.0.255 192.18.50.0 0.0.0.255
Router(config)# access-list 188 permit ip any any
                                  or
          access-list 188 permit ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
Router(config)# interface e0
Router(config-if)# ip access-group 188 in
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```

[Disabling ACL's]

```
Router# configure terminal
Router(config)# interface e0
Router(config-if)# no ip access-group 188 out
Router(config-if)# exit
Router(config)# exit
```

[Removing an ACL]

```
Router# configure terminal
Router(config)# interface e0
Router(config-if)# no ip access-group 188 out
Router(config-if)# exit
Router(config)# no access-list 188
Router(config)# exit
```



Extended Access List Problem #5 — Deny/Permit Entire Ranges

Write an extended access list to permit network 204.95.150.0 to send packets to network 172.59.0.0, but not the 210.250.10.0 network. Permit all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: FA1
Access-list #: 125 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)#access-list 125 deny ip 204.95.150.0 0.0.0.255 210.250.10.0 0.0.0.255
              access-list 125 permit ip any any
Router(config)# interface FA0
Router(config-if)# ip access-group 125 (in) or out (circle one)
Router(config-if)# exit
Router(config)# exit
```

Extended Access List Problem #6 — Deny/Permit Entire Ranges

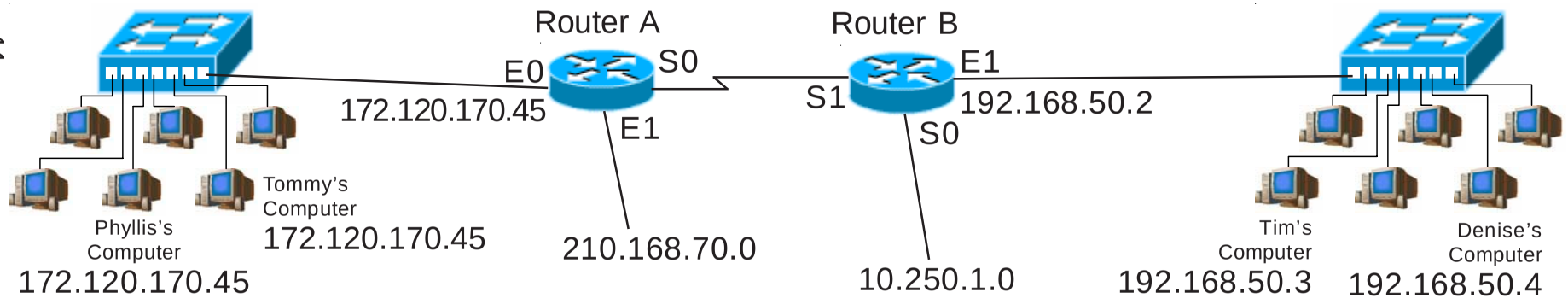
Write an extended access list to allow Rachel's computer at 204.95.150.10 to receive information from the 172.59.0.0 network. Deny all other hosts on the 204.95.150.0 network access from the 172.59.2.0 network. Permit all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: FA1
Access-list #: 130 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal
Router(config)# access-list 130 permit ip 172.59.0.0 0.0.255.255 host 204.95.150.10
          or
               access-list 130 permit ip 172.59.0.0 0.0.255.255 204.95.150.10 0.0.0.0
               access-list 130 deny ip 172.59.0.0 0.0.255.255 204.95.150.0 0.0.0255
               access-list 130 permit any any
```

```
Router(config)# interface FA1
Router(config-if)# ip access-group 130 (in) or out (circle one)
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```



Extended Access List Problem #7 — Deny/Permit Entire Ranges

Write a named extended access list called "Godzilla" to prevent the 172.120.0.0 network from sending information to the 210.168.70.0 , and 10.250.1.0 255.255.255.0 networks; but will permit traffic to the 192.168.50.0 network. Permit all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: E0
Access-list Name: Godzilla

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)#access-list extended Godzilla
Router(config-ext-nacl)# deny ip 172120.0.0 0.0.255.255 210.168.70.0 0.0.0.255
                          deny ip 172.120.0.0 0.0.255.255 10.250.1.0 0.0.0.255
                          permit ip any any
Router(config-ext-nacl)# interface E0
Router(config-if)# ip access-group Godzilla in or out (circle one)
Router(config-if)# exit
Router(config)# exit
```

Extended Access List Problem #8 — Deny/Permit Entire Ranges

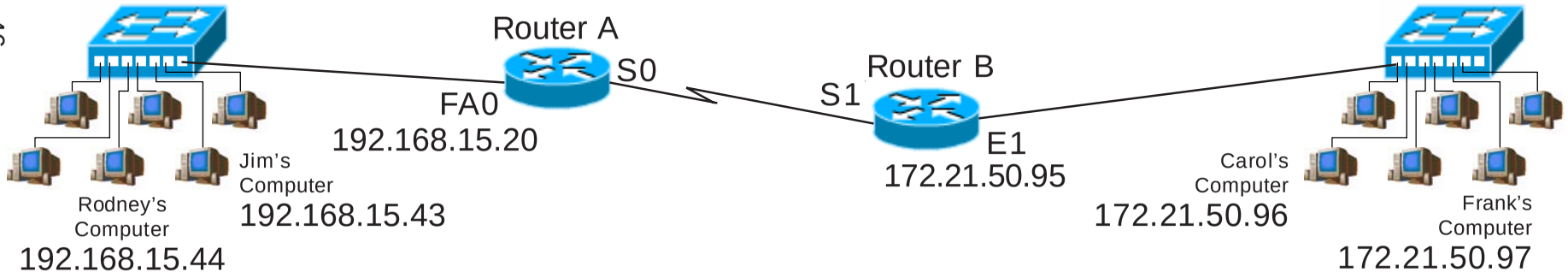
Assuming default subnet masks write an extended access list to permit Tim at 192.168.50.3 to receive data from the 172.120.0.0 network. Allow the 192.168.50.0 network to receive information from Phyllis's computer at 172.120.170.45. Deny all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: E0
Access-list #: 140 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal
Router(config)# access-list 140 permit ip 172.120.0.0 0.0.255.255 host 192.168.50.3
        or access-list 140 permit ip 172.120.0.0 0.0.255.255 192.168.50.3 0.0.0.0
           access-list 140 permit ip host 172.120.170.45 192.168.50.0 0.0.0.255
        or access-list 140 permit ip 172.120.170.45 0.0.0.0 192.168.50.0 0.0.0.255
```

```
Router(config)# interface E0
Router(config-if)# ip access-group 140 (in) or out (circle one)
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```



Extended Access List Sample #5 — Deny/Permit a Range of Addresses

Write an extended access list to deny the first 15 usable addresses of the 192.168.15.0 network from reaching the 172.21.0.0 network. Permit all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: FA0
Access-list #: 185

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)# access-list 185 deny ip 192.168.15.0 0.0.0.15 172.21.50.0 0.0.255.255
Router(config)# access-list 185 permit ip any any
                              or
                access-list 185 permit ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
Router(config)# interface fa1
Router(config-if)# ip access-group 185 in
Router(config-if)# exit
Router(config)# exit
```

[Viewing information about existing ACL's]

`Router# show configuration`	(This will show which access groups are associated with particular interfaces)
`Router# show access list 185`	(This will show detailed information about this ACL)

Extended Access List Sample #6 — Deny/Permit a Range of Addresses

Write an extended access list which will allow the lower half of 192.168.15.0 network access to the 172.21.50.0 network. Deny all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: FA0
Access-list #: 121

[Writing and installing an ACL]

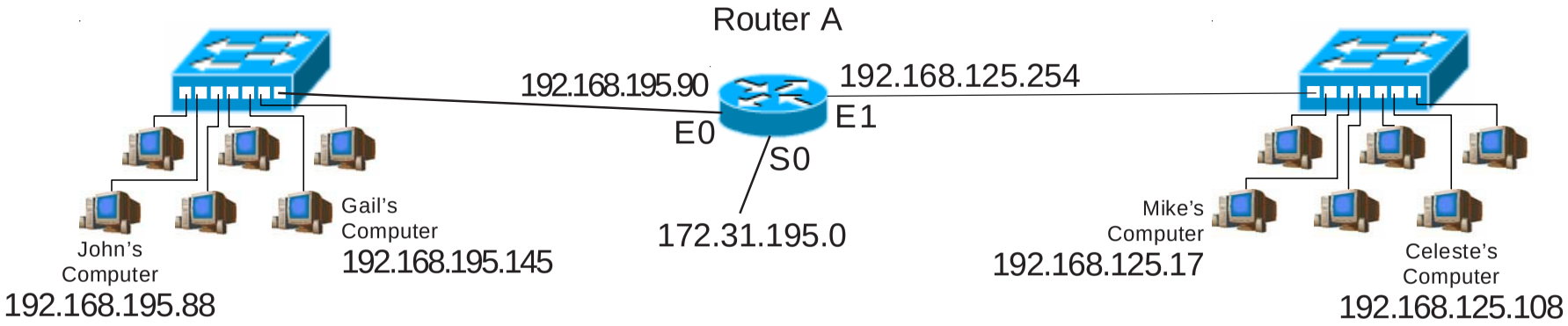
```
Router# configure terminal
Router(config)# access-list 121 permit ip 192.168.15.0 0.0.0.127 172.21.50.0 0.0.0.255
Router(config)# access-list 121 deny ip any any
                                  or
                access-list 121 deny ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
Router(config)# interface fa0
Router(config-if)# ip access-group 121 in
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```

[Disabling ACL's]

```
Router# configure terminal
Router(config)# interface fa0
Router(config-if)# no ip access-group 121 in
Router(config-if)# exit
Router(config)# exit
```

[Removing an ACL]

```
Router# configure terminal
Router(config)# interface fa0
Router(config-if)# no ip access-group 121 in
Router(config-if)# exit
Router(config)# no access-list 121
Router(config)# exit
```



Extended Access List Problem #9 — Deny/Permit a Range of Addresses

Write an extended access list to prevent the first 31 usable addresses in the 192.168.125.0 network from reaching the 192.168.195.0 network. Permit all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: E1
Access-list #: 145 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal (or config t)

Router(config)#access-list 145 deny ip 192.168.125.0 0.0.0.31 192.168.195.0 0.0.0.255
              access-list 145 permit ip any any

Router(config)# interface E1
Router(config-if)# ip access-group 145 (in) or out (circle one)
Router(config-if)# exit
```

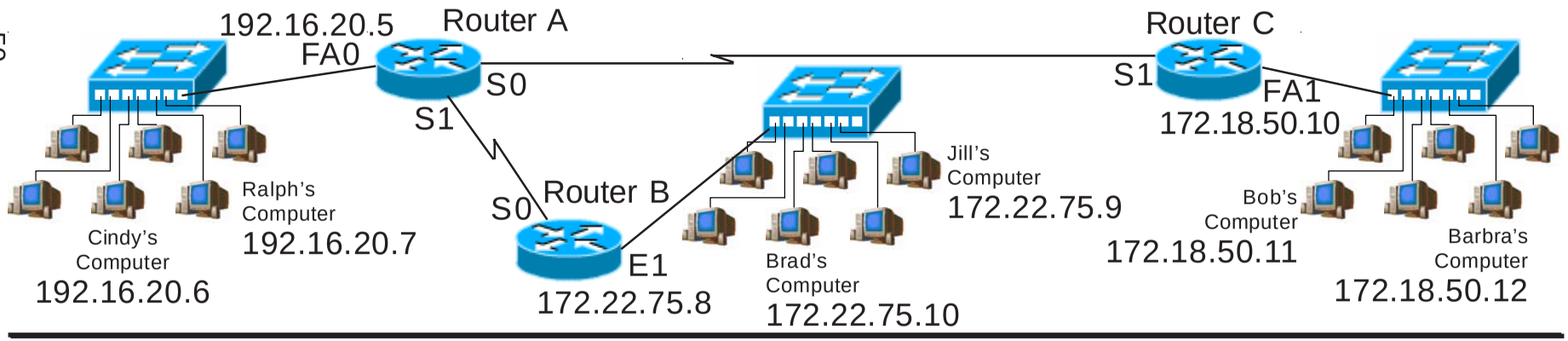
Extended Access List Problem #10 — Deny/Permit a Range of Addresses

Write a named extended access list called "Media_Center" to permit the range of addresses from 172.31.195.1 through 172.31.195.7 to send date to the 192.168.125.0 network. Deny all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: S0
Access-list Name: Media_Center

[Writing and installing an ACL]

```
Router# configure terminal
Router(config)# access-list extended Media_Center
Router(config-ext-nacl)# permit ip 172.31.195.0 0.0.0.7 192.168.125.0 0.0.0.255
Router(config-ext-nacl)# interface S0
Router(config-if)# ip access-group Media_Center in or out (circle one)
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```



Extended Access List Problem #11 — Deny/Permit a Range of Addresses

Write an extended access list to permit the first 3 usable addresses in the 192.16.20.0 network to reach the 172.22.75.0 network. Deny the addresses from 192.16.20.4 through 192.16.20.31 from reaching the 172.22.75.0 network. Permit all other traffic. Keep in mind that there are multiple ways this ACL can be written.

Place the access list at:
Router Name: Router A
Interface: FA0
Access-list #: 155 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)#access-list 155 permit ip 192.16.20.0 0.0.0.3 172.22.75.0 0.0.0.255
              access-list 155 deny ip 192.16.2.0 0.0.0.31 172.22.75.0 0.0.0.255
              access-list 155 permit ip any any

Router(config)# interface FA0
Router(config-if)# ip access-group 155 (in) or out (circle one)
Router(config-if)# exit
```

Extended Access List Problem #12 — Deny/Permit a Range of Addresses

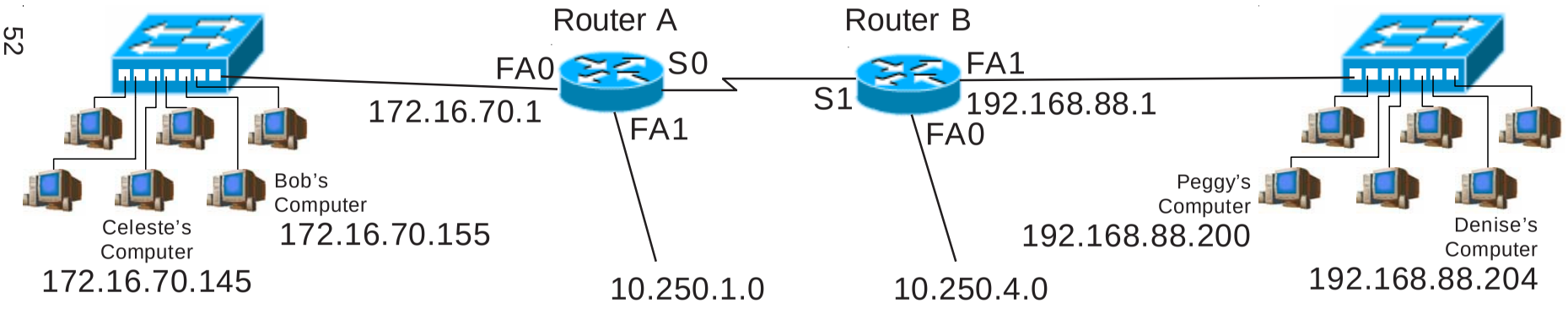
Write an extended access list to deny the addresses from 172.22.75.8 through 172.22.75.127 from sending data to the 172.18.50.0 network. Deny the first half of the addresses from the 172.22.75.0 network from reaching the 192.16.20.0 network. Permit all other traffic. Keep in mind that there are multiple ways this ACL can be written.

Place the access list at:
Router Name: Router B
Interface: E1
Access-list #: 160 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal
Router(config)# access-list 160 permit ip 172.22.75.0 0.0.0.7 172.18.50.0 0.0.0.255
                access-list 160 deny ip 172.22.75.0 0.0.0.127 172.18.50.0 0.0.0.255
                access-list 160 permit ip any any
```

```
Router(config)# interface E1
Router(config-if)# ip access-group 160 (in) or out (circle one)
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```



Extended Access List Problem #13 — Deny/Permit a Range of Addresses

Write an extended access list to permit the first 63 usable addresses in the 192.168.88.0 network to reach the lower half of the addresses in the 172.16.70.0 network; but not the upper half. Deny all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: FA1
Access-list #: 165 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)#access-list 165 permit ip 192.168.88.0 0.0.0.63 172.16.70.0 0.0.0.127
Router(config)# interface FA1
Router(config-if)# ip access-group 165 (in) or out (circle one)
Router(config-if)# exit
```

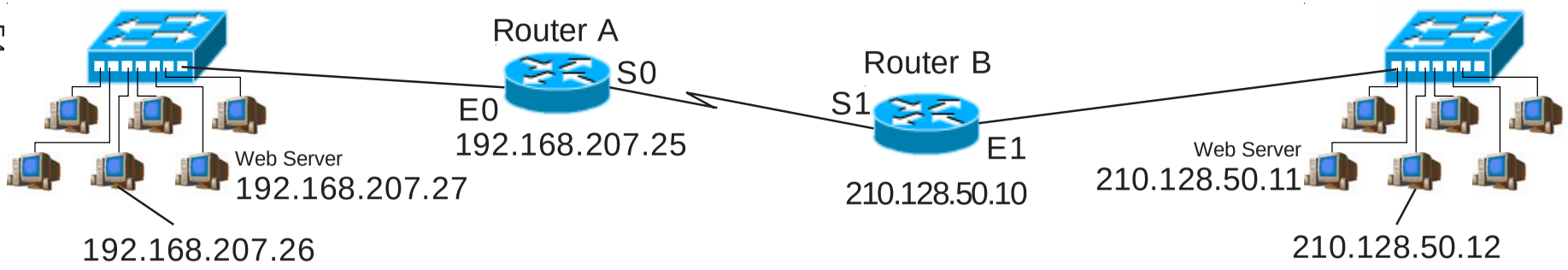
Extended Access List Problem #14 — Deny/Permit a Range of Addresses

Write an extended access list to deny the addresses from 10.250.1.0 through 10.250.1.63 from sending data to Denise's computer. Permit all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: FA1
Access-list #: 170 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal
Router(config)# access-list 170 deny ip 10.250.1.0 0.0.0.63 host 192.168.88.204
        or  access-list 170 deny ip 10.250.1.0 0.0.0.63 192.168.88.204 0.0.0.0
            access-list 170 permit ip any any
Router(config)# interface FA1
Router(config-if)# ip access-group 170 (in) or out (circle one)
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```



Extended Access List Sample #7 — Deny/Permit Port Numbers

Write an extended access list to deny HTTP traffic intended for web server 192.168.207.27, but will permit all other HTTP traffic to reach the only the 192.168.207.0 network. Deny all other IP traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: E0
Access-list #: 198

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)# access-list 198 deny tcp any 192.168.207.27 0.0.0.0 eq www
                                  or
                access-list 198 deny tcp any host 192.168.207.27 eq www
Router(config)# access-list 198 permit tcp any 192.168.207.0 0.0.0.255 eq www
Router(config)# interface e0
Router(config-if)# ip access-group 198 in
Router(config-if)# exit
Router(config)# exit
```

[Viewing information about existing ACL's]

`Router# show configuration`	(This will show which access groups are associated with particular interfaces)
`Router# show access list 198`	(This will show detailed information about this ACL)

Extended Access List Sample #8 — Deny/Permit Port Numbers

Write an extended access list to permit pings in either direction between hosts on the 210.128.50.0 and 192.168.207.0 networks. Deny all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: E0
Access-list #: 134

[Writing and installing an ACL]

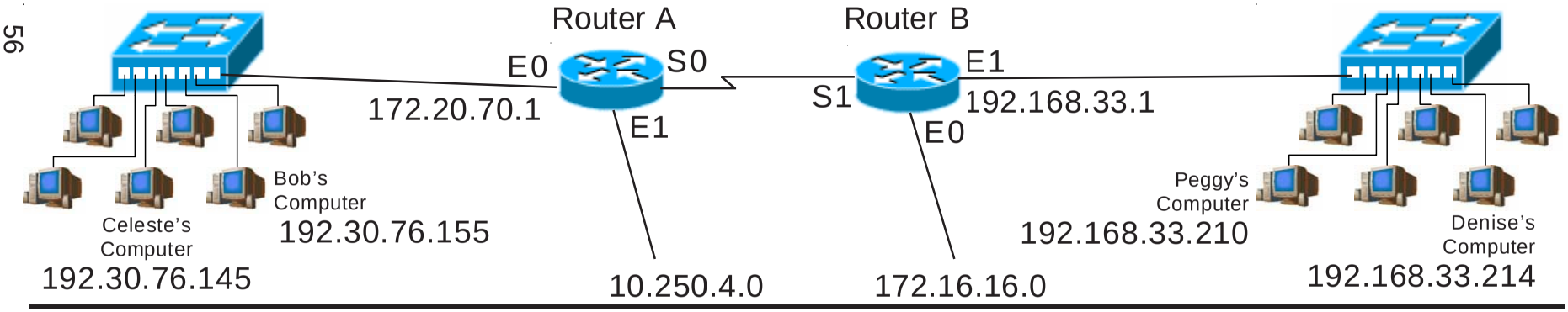
```
Router# configure terminal
Router(config)# access-list 134 permit icmp 210.128.50.0 0.0.0.255 192.168.207.0 0.0.0.255 echo-reply
Router(config)# interface e0
Router(config-if)# ip access-group 134 in
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```

[Disabling ACL's]

```
Router# configure terminal
Router(config)# interface e0
Router(config-if)# no ip access-group 134 out
Router(config-if)# exit
Router(config)# exit
```

[Removing an ACL]

```
Router# configure terminal
Router(config)# interface e0
Router(config-if)# no ip access-group 134 out
Router(config-if)# exit
Router(config)# no access-list 134
Router(config)# exit
```



Standard Access List Sample #9 — Deny/Permit Telnet

Write an extended access list to permit Denise's and Bob's computers to telnet into Router B. Deny all other telnet traffic Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: line VTY 0 4 (using line VTY 0 4 instead of an interface like E1 allows you to apply this access list to all VTY lines with one statement)
Access-list #: 45

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)# access-list 45 permit 192.168.33.214 0.0.0.0
                                    or
                access-list 45 permit host 192.168.33.214
Router(config)# access-list 45 permit 192.30.76.155 0.0.0.0
                                    or
                access-list 45 permit host 92.30.76.155
Router(config)# line vty 0 4
Router(config-if)# ip access-class 45 in
Router(config-if)# exit
Router(config)# exit
```

[Viewing information about existing ACL's]

`Router# show configuration`	(This will show which access groups are associated with particular interfaces)
`Router# show access list 45`	(This will show detailed information about this ACL)

Extended Access List Sample #10 — Deny/Permit Port Numbers

Write an extended access list to deny FTP to ip addresses 192.30.76.0 through 192.30.76.13.
Permit all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: E0
Access-list #: 155

[Writing and installing an ACL]

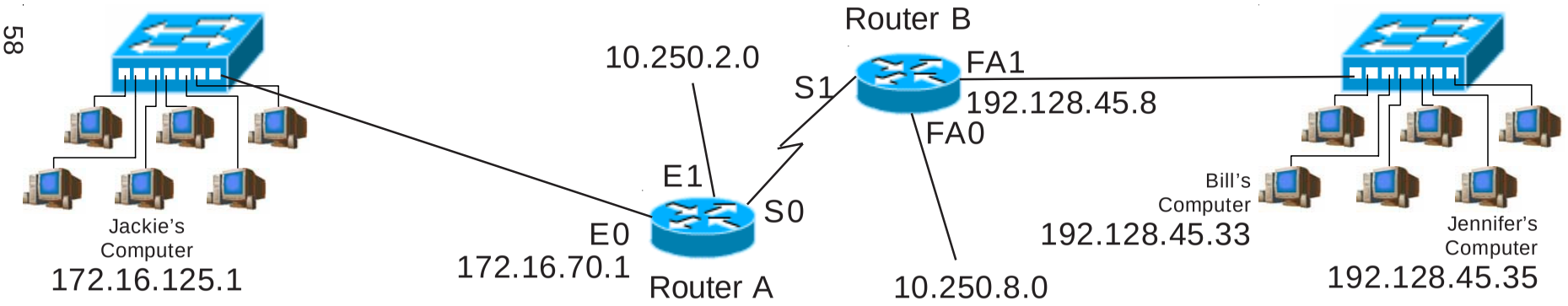
```
Router# configure terminal
Router(config)# access-list 155 deny tcp any 192.30.76.0 0.0.0.13 eq ftp
Router(config)# access-list 155 permit tcp any any
                              or
                  access-list 155 deny tcp 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
Router(config)# interface e0
Router(config-if)# ip access-group 155 in
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```

[Disabling ACL's]

```
Router# configure terminal
Router(config)# interface e0
Router(config-if)# no ip access-group 155 out
Router(config-if)# exit
Router(config)# exit
```

[Removing an ACL]

```
Router# configure terminal
Router(config)# interface e0
Router(config-if)# no ip access-group 155 out
Router(config-if)# exit
Router(config)# no access-list 155
Router(config)# exit
```



Extended Access List Problem #15 — Deny/Permit a Port Numbers

Write an extended access list to permit ICMP traffic from the 192.128.45.0 network to reach the 172.16.125.0 255.255.255.0 and 10.250.2.0 255.255.255.0 networks. Deny all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: FA1
Access-list #: 175 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)#access-list 175 permit icmp 192.128.45.0 0.0.0.255 172.16.125.0 0.0.0.255
              access-list 175 permit icmp 192.128.45.0 0.0.0.255 10.250.2.0 0.0.0.255
Router(config)# interface FA1
Router(config-if)# ip access-group 175 (in) or out (circle one)
Router(config-if)# exit
```

Extended Access List Problem #16 — Deny/Permit a Port Numbers

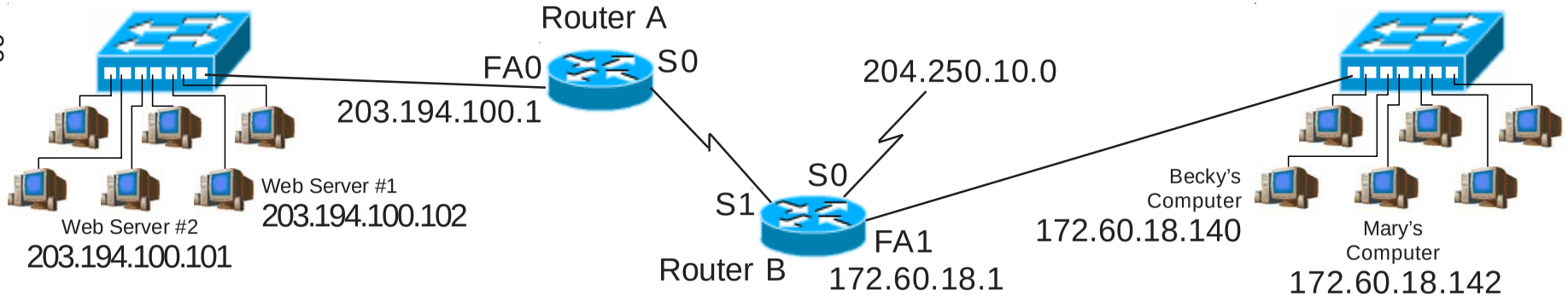
Write a named extended access list called "Peggys_Lab" to deny telnet from 10.250.8.0 through 10.250.8.127 from reaching the 192.128.45.0 network. Permit all other traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: FA0
Access-list Name: Peggys_Lab

[Writing and installing an ACL]

```
Router# configure terminal
Router(config)# access-list extended Peggys_Lab
Router(config-ext-nacl deny tcp 10.250.8.0 0.0.0.127 192.128.45.0 0.0.0.255 eq 23
                       permit tcp any any
```

```
Router(config-ext-nacl)# interface FA0
Router(config-if)# ip access-group Peggys_Lab (in) or out (circle one)
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```



Access List Problem #17 — Deny/Permit Port Numbers

Write an access list to permit Becky and Mary's computer to telnet into Router B. Deny all other telnet traffic from the 172.60.18.0 network. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: line vty 04
Access-list #: 50 (1-99)

[Writing and installing an ACL]

```
Router# configure terminal (or config t)

Router(config)# access-list 50 permit 172.60.18.140
            or access-list 50 permit host 172.60.18.140
            or access-list 50 permit 172.60.18.140 0.0.0.0
               access-list 50 permit 172.60.18.142
            or access-list 50 permit host 172.60.18.142
            or access-list 50 permit 172.60.18.142 0.0.0.0

Router(config)# interface line vty 04
Router(config-if)# ip access-group 50 (in) or out (circle one)
Router(config-if)# exit
Router(config)# exit
```

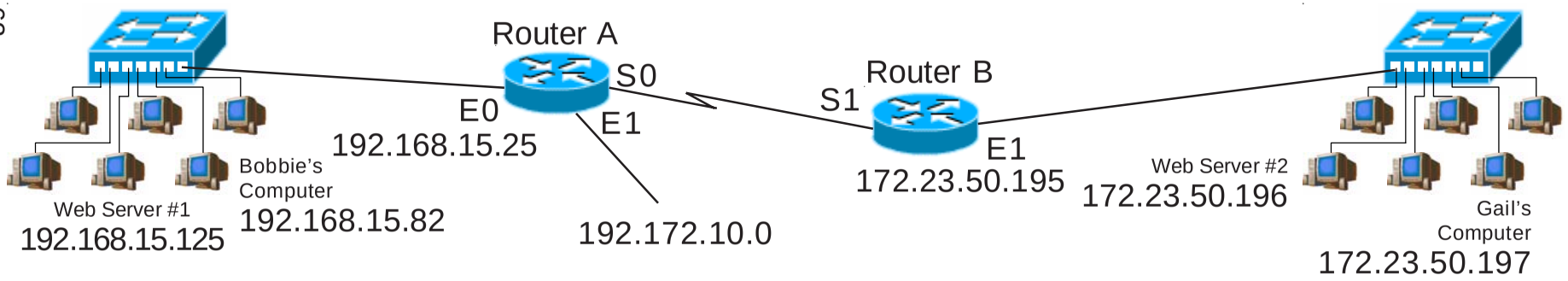
Extended Access List Problem #18 — Deny/Permit Port Numbers

Write an extended access list to deny all HTTP traffic intended for the web server at 203.194.100.102. Permit HTTP traffic to any other web servers. Deny all other IP traffic to the 203.194.100.0 network. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: FA0
Access-list #: 185 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)#   access-list 185 deny tcp any host 203.194.100.102 eq 80
                  or
                  access-list 185 deny tcp any 203.194.100.102 0.0.0.0 eq 80
                  access-list 185 permit tcp any any eq 80
Router(config)# interface FA0
Router(config-if)# ip access-group 185 in or out (circle one)
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```



Access List Problem #19 — Deny/Permit Port Numbers

Write an access list to permit TFTP traffic to all hosts on the 192.168.15.0 network. Deny all other TFTP traffic. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router A
Interface: E0
Access-list #: 190 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal (or config t)
Router(config)#access-list 175 permit tcp any 192.168.15.0 0.0.0.255 eq ftp
Router(config)# interface E0
Router(config-if)# ip access-group 190 (in) or out (circle one)
Router(config-if)# exit
Router(config)# exit
```

Extended Access List Problem #20 — Deny/Permit Port Numbers

Write an extended access list that permits web traffic from web server #2 at 172.23.50.196 to reach everyone on the 192.168.15.0 network. Deny all other IP traffic going to the 192.172.10.0, and 192.168.15.0 networks. Keep in mind that there may be multiple ways many of the individual statements in an ACL can be written.

Place the access list at:
Router Name: Router B
Interface: E1
Access-list #: 195 (100-199)

[Writing and installing an ACL]

```
Router# configure terminal
Router(config)# access-list 195 deny tcp host 172.23.50.196 192.168.15.0 0.0.0.255 eq 80
        or
               access-list 195 deny tcp 172.23.50.196 0.0.0.0 192.168.15.0 0.0.0.255 eq 80
```

```
Router(config)# interface E1
Router(config-if)# ip access-group 195 (in) or out (circle one)
Router(config-if)# exit
Router(config)# exit
Router# copy run start
```

Optional ACL Commands

& Other Network Security Ideas

In order to reduce the chance of spoofing from outside your network consider adding the following statements to your network's inbound access list.

```
router# config t
router(config)# access-list 100 deny ip 10.0.0.0 0.255.255.255 any
router(config)# access-list 100 deny ip 172.16.0.0 0.0.255.255 any
router(config)# access-list 100 deny ip 192.168.0.0 0.0.255.255 any
router(config)# access-list 100 deny ip 127.0.0.0 0.255.255.255 any
router(config)# access-list 100 deny ip 224.0.0.0 31.255.255.255 any
router(config)# access-list 100 deny ip your-subnet-#  your-subnet-mask-# any
router(config)# access-list 100 deny igmp any any
router(config)# access-list 100 deny icmp any any redirect
router(config)# access-list 100 permit any any
router(config)# interface e0 (or whatever your inbound port is)
router(config-if)# ip access-group in
router(config-if)# exit
router(config)# exit
```

Another handy security tool is to only allow ip packets out of your network with your source address.

```
router# config t
router(config)# access-list 100 permit ip your-subnet-#  your-subnet-mask-# any
router(config)# interface e0 (or whatever your outbound port is)
router(config-if)# ip access-group out
router(config-if)# exit
router(config)# exit
```

To keep packets with unreachable destinations from entering your network add this command:

```
ip route 0.0.0.0 0.0.0.0 null 0 255
```

To protect against smurf and other attacks add the following commands to every external interface:

```
no ip directed-broadcast
no ip source-route
fair-queue
scheduler interval 500
```

Index / Table of Contents

Port Numbers

Port numbers are now assigned by the ICANN (Internet Corporation for Assigned Names and Numbers). Commonly used TCP and UDP applications are assigned a port number; such as: HTTP - 80, POP3 - 110, FTP - 20. When an application communicates with another application on another node on the internet, it specifies that application in each data transmission by using its port number. You can also type the name (ie. Telnet) instead of the port number (ie. 23). Port numbers range from 0 to 65536 and are divided into three ranges:

Well Known Ports	0	to	1,023
Registered Ports	1,024	to	49,151
Dynamic and/or Private Ports	49,152	to	65,535

Below is a short list of some commonly used ports. For a complete list of port numbers go to http://www.iana.org/assignments/port-numbers.

Some commonly used port numbers:

0	Reserved	
1	TCPMUX	(TCP Port Service Multiplexer)
5	RJE	(Remote Job Entry)
7	ECHO	
9	DISCARD	
11	SYSTAT	(Active users)
13	DAYTIME	
17	QUOTE	(Quote of the day)
18	MSP	(Message Send Protocol)
19	CHARGEN	(Character generator)
20	FTP-DATA	(File Transfer Protocol - Data)
21	FTP	(File Transfer Protocol - Control)
22	SSH	(Remote Login Protocol)
23	Telnet	(Terminal Connection)
25	SMTP	(Simple Mail Transfer Protocol)
29	MSG ICP	
37	TIME	
39	RLP	(Resource Location Protocol
42	NAMESERV	(Host Name Server)

43	NICNAME	(Who Is)
49	LOGIN	(Login Host Protocol)
53	DNS	(Domain Name Server)
67	BOOTP	(Bootstrap Protocol Server)
68	BOOTPS	(Bootstrap Protocol Client)
69	TFTP	(Trivial File Transfer Protocol)
70	GOPHER	(Gopher Services)
75		(Any Privite Dial-out Service)
79	FINGER	
80	HTTP	(Hypertext Transfer Protocol)
95	SUPDUP	(SUPDUP Protocol)
101	HOSTNAME	(NIC Host Name Server)
108	SNAGAS	(SNA Gateway Access Server)
109	POP2	(Post Office Protocol - Version 2)
110	POP3	(Post Office Protocol - Version 3)
113	AUTH	(Authentication Service)
115	SFTP	(Simple File Transfer Protocol)
117	UUCP-PATH	(UUCP Path Service)
118	SQLSERV	(SQL Services)
119	NNTP	(Newsgroup)
123	NTP	(Network Tim Protocol)
137	NetBIOS-NS	(NetBIOS Name Service)
139	NetBIOS-SSN	(NetBIOS Session Service)
143	IMAP	(Interim Mail Access Protocol)
150	SQL-NET	(NetBIOS Session Service)
156	SQLSRV	(SQL Service)
161	SNMP	(Simple Network Management Protocol)
179	BGP	(Border Gateway Protocol)
190	GACP	(Gateway Access Control Protocol)
194	IRC	(Internet Relay Chat)
197	DLS	(Directory Location Service)
389	LDAP	(Lightweight Directory Access Protocol)
396	NETWARE-IP	(Novell Netware over IP)
443	HTTPS	(HTTP MCom)
444	SNPP	(Simple Network Paging Protocol)
445	Microsoft-DS	
458	Apple QuickTime	
546	DHCP Client	
547	DHCP Server	
563	SNEWS	
569	MSN	