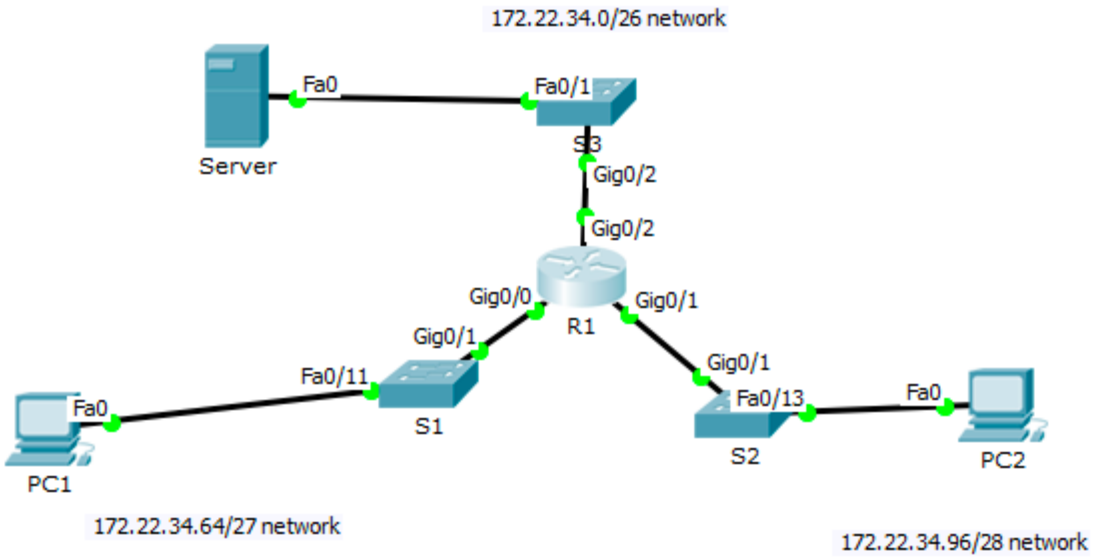
TASK 2 - Configuring Extended ACLs

Note: Red font color INDICATES WHERE YOU WILL NEED TO CONFIGURE / CALCULATE VALUES.

Topology



IMPORTANT: For each of your routers/switches hostnames insert your student number. E.g. R1 – 2015111R1. Read the instructions carefully and complete the task as required.

Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0	172.22.34.65	255.255.255.224	N/A
	G0/1	172.22.34.97	255.255.255.240	N/A
	G0/2	172.22.34.1	255.255.255.192	N/A
Server	NIC	172.22.34.62	255.255.255.192	172.22.34.1
PC1	NIC	172.22.34.66	255.255.255.224	172.22.34.65
PC2	NIC	172.22.34.98	255.255.255.240	172.22.34.97

Equipment to choose in Packet Tracer:

- **Router 2911 (this router provides 3 x Gig ports)**
- **Switch 2960 (these switches will provide you with 2 x Gig ports)**
- **Generic Server**

Objectives

Part 1:Configure all interfaces and test connectivity. Configure, Apply and Verify an Extended Numbered ACL

Part 2: Configure, Apply and Verify an Extended Named ACL

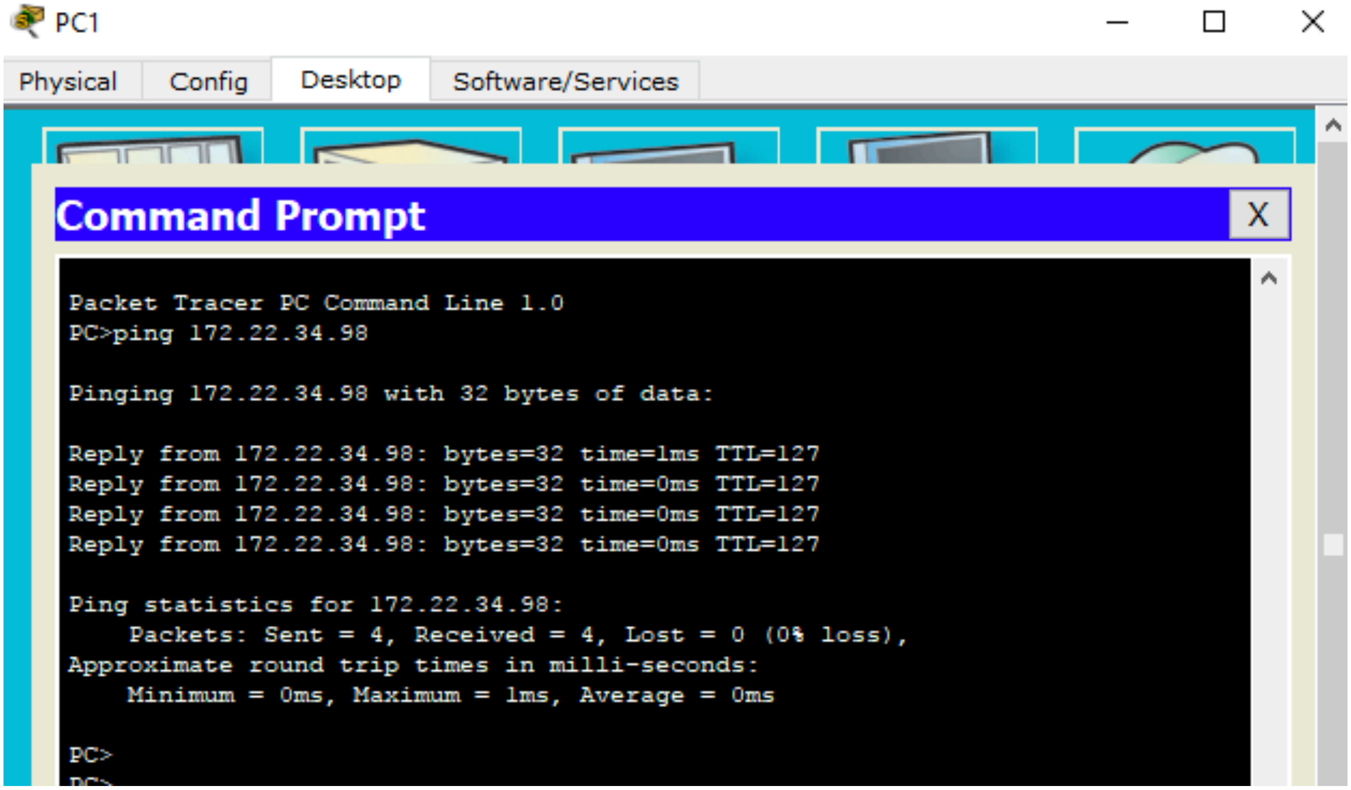
Background / Scenario

Two employees need access to services provided by the server. **PC1** only needs FTP access while **PC2** only needs web access. Both computers are able to ping the server, but not each other.

Part 1: Configure all interfaces and test connectivity. Configure, Apply and Verify an Extended Numbered ACL

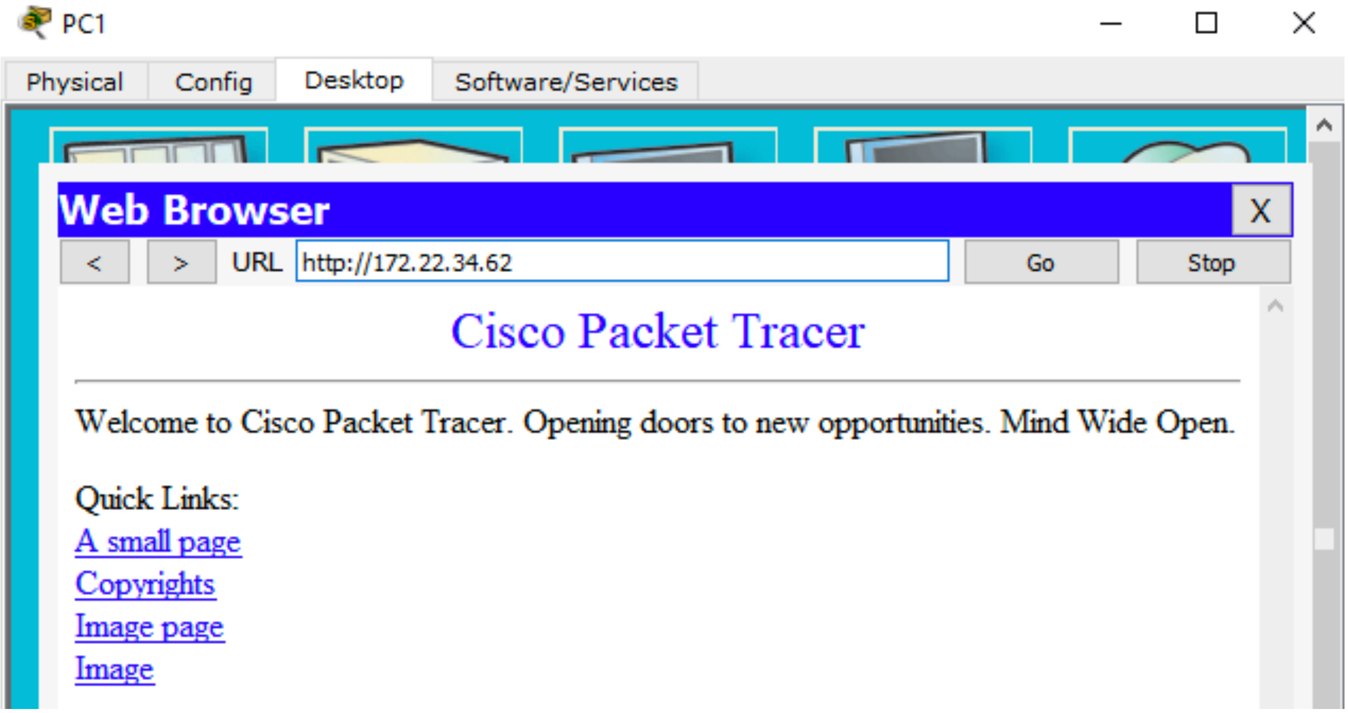
Configure all interfaces as shown in the table above. **Ensure all devices can ping each other** e.g. PC1 can ping Server, PC2 can ping Server etc etc. **If they cannot ping ensure to check IP configuration.**

As shown below PC1 can ping PC2.



```
Packet Tracer PC Command Line 1.0
PC>ping 172.22.34.98

Pinging 172.22.34.98 with 32 bytes of data:

Reply from 172.22.34.98: bytes=32 time=1ms TTL=127
Reply from 172.22.34.98: bytes=32 time=0ms TTL=127
Reply from 172.22.34.98: bytes=32 time=0ms TTL=127
Reply from 172.22.34.98: bytes=32 time=0ms TTL=127

Ping statistics for 172.22.34.98:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>
PC>
```

PC1 can also open a Web browser and access the web server's homepage.



http://172.22.34.62

Cisco Packet Tracer

Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open.

Quick Links:
A small page
Copyrights
Image page
Image

Step 1: Configure an ACL to permit FTP and ICMP ONLY for LAN where PC1 resides.

a. From global configuration mode on **R1**, enter the following command to determine the first valid number for an extended access list.

```
R1(config)# access-list ?
  <1-99>     IP standard access list
  <100-199>  IP extended access list
```

b. Add **100** to the command, followed by a question mark.

```
R1(config)# access-list 100 ?
  deny     Specify packets to reject
  permit   Specify packets to forward
  remark   Access list entry comment
```

c. To permit FTP traffic, enter **permit,** followed by a question mark.

```
R1(config)# access-list 100 permit ?
  ahp    Authentication Header Protocol
  eigrp  Cisco's EIGRP routing protocol
  esp    Encapsulation Security Payload
  gre    Cisco's GRE tunneling
  icmp   Internet Control Message Protocol
  ip     Any Internet Protocol
  ospf   OSPF routing protocol
  tcp    Transmission Control Protocol
  udp    User Datagram Protocol
```

Task 2 - Configuring Extended ACLs

d. This ACL permits FTP and ICMP. ICMP is listed above, but FTP is not, because FTP uses TCP. So you enter TCP. Enter **tcp** to further refine the ACL help.

```
R1(config)# access-list 100 permit tcp ?
  A.B.C.D  Source address
  any      Any source host
  host     A single source host
```

e. Notice that we could filter just for **PC1** by using the **host** keyword or we could allow **any** host. In our case, we want to allow any device belonging to the **172.22.34.64/27** network. Enter the network address, followed by THE WILDCARD MASK. You will need to calculate this wildcard mask for this network. You will enter the wildcard mask where it shows the ? below.

```
R1(config)# access-list 100 permit tcp 172.22.34.64 ?
```

f. Calculate the wildcard mask determining the binary opposite of a subnet mask.

```
11111111.11111111.11111111.11100000 = 255.255.255.224
00000000.00000000.00000000.00011111 = ?.?.?.?
```

g. Enter the wildcard mask, followed by a question mark.

```
R1(config)# access-list 100 permit tcp 172.22.34.64 [Wildcard mask here] ?
  A.B.C.D  Destination address
  any      Any destination host
  eq       Match only packets on a given port number
  gt       Match only packets with a greater port number
  host     A single destination host
  lt       Match only packets with a lower port number
  neq      Match only packets not on a given port number
  range    Match only packets in the range of port numbers
```

h. Configure the destination address. In this scenario, we are filtering traffic for a single destination, the server. Enter the **host** keyword followed by the server's IP address.

```
R1(config)# access-list 100 permit tcp 172.22.34.64 [Wildcard mask here] host
172.22.34.62 ?
  dscp         Match packets with given dscp value
  eq           Match only packets on a given port number
  established  established
  gt           Match only packets with a greater port number
  lt           Match only packets with a lower port number
  neq          Match only packets not on a given port number
  precedence   Match packets with given precedence value
  range        Match only packets in the range of port numbers
  <cr>
```

i. Notice that one of the options is **<cr>** (carriage return). In other words, you can press **Enter** and the statement would permit all TCP traffic. However, we are only permitting FTP traffic; therefore, enter the **eq** keyword, followed by a question mark to display the available options. Then, enter **ftp** and press **Enter**.

```
R1(config)# access-list 100 permit tcp 172.22.34.64 [Wildcard mask here] host
172.22.34.62 eq ?
  <0-65535>  Port number
```

```
    ftp         File Transfer Protocol (21)
    pop3        Post Office Protocol v3 (110)
    smtp        Simple Mail Transport Protocol (25)
    telnet      Telnet (23)
    www         World Wide Web (HTTP, 80)
R1(config)# access-list 100 permit tcp 172.22.34.64 [Wildcard mask here] host
172.22.34.62 eq ftp
```

j. Create a second access list statement to permit ICMP (ping, etc.) traffic from **PC1 LAN** to **Server**. Note that the access list number remains the same. The wildcard mask again will be for the whole subnet of 172.22.34.64/27.

```
R1(config)# access-list 100 permit icmp 172.22.34.64 [Wildcard mask here]
host 172.22.34.62
```

k. All other traffic is denied, by default.

Step 2: Apply the ACL on the correct interface to filter traffic.

From **R1**’s perspective, the traffic that ACL 100 applies to is **inbound** from the network connected to **Gigabit Ethernet 0/0 interface**. Enter interface configuration mode and apply the ACL.

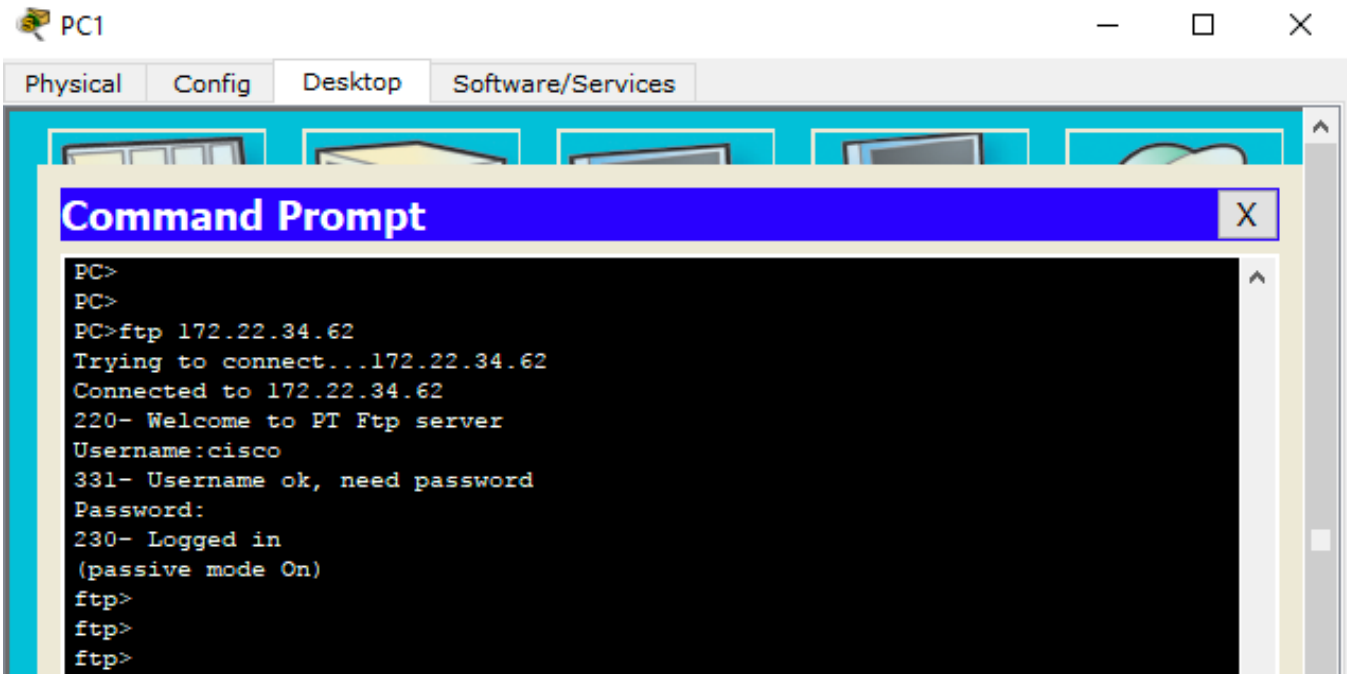
```
R1(config)# interface gigabitEthernet [interface port number here]
R1(config-if)# ip access-group 100 [direction]
```

Step 3: Verify the ACL implementation.

a. Ping from **PC1** to **Server**. If the pings are unsuccessful, verify the IP addresses before continuing.

b. FTP from **PC1** to **Server**. The username and password are both **cisco**. This should allow you to connect. Enter into the command prompt on PC1 and do the following:

```
PC> ftp 172.22.34.62
```

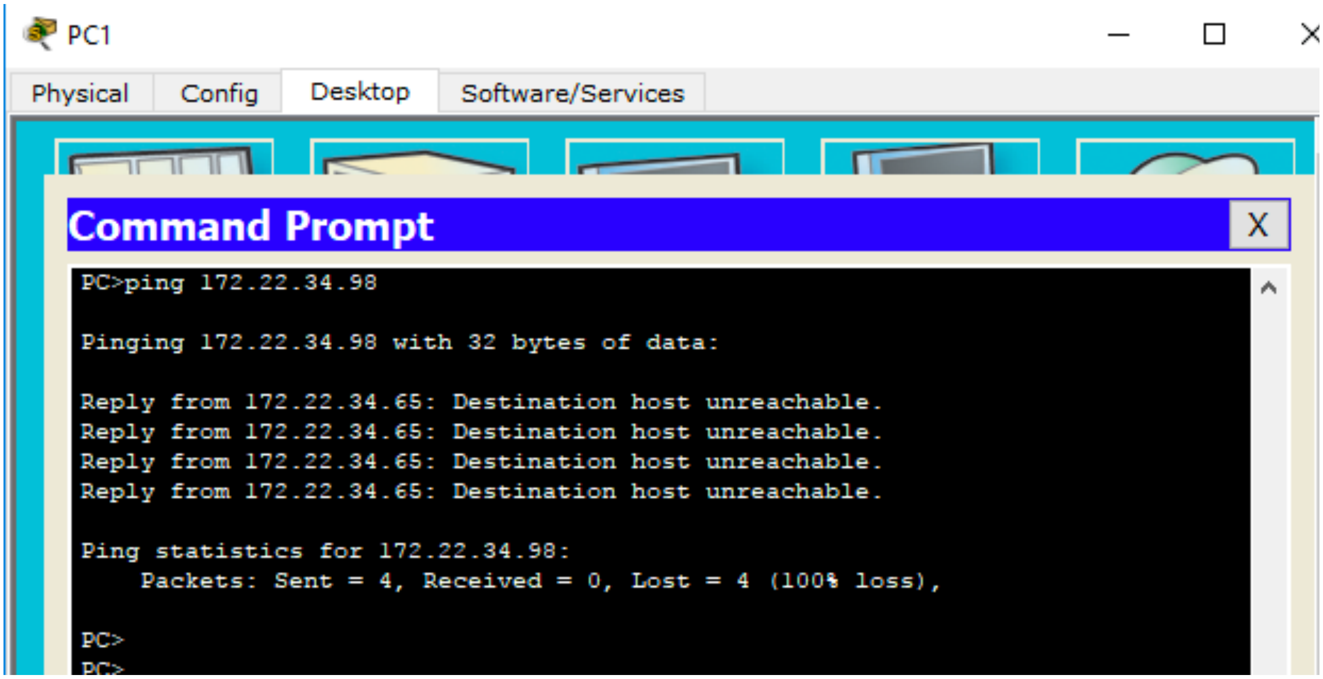


```
PC>
PC>
PC>ftp 172.22.34.62
Trying to connect...172.22.34.62
Connected to 172.22.34.62
220- Welcome to PT Ftp server
Username:cisco
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>
ftp>
ftp>
```

c. Exit the FTP service of the **Server**. This proves you can connect using the FTP protocol.

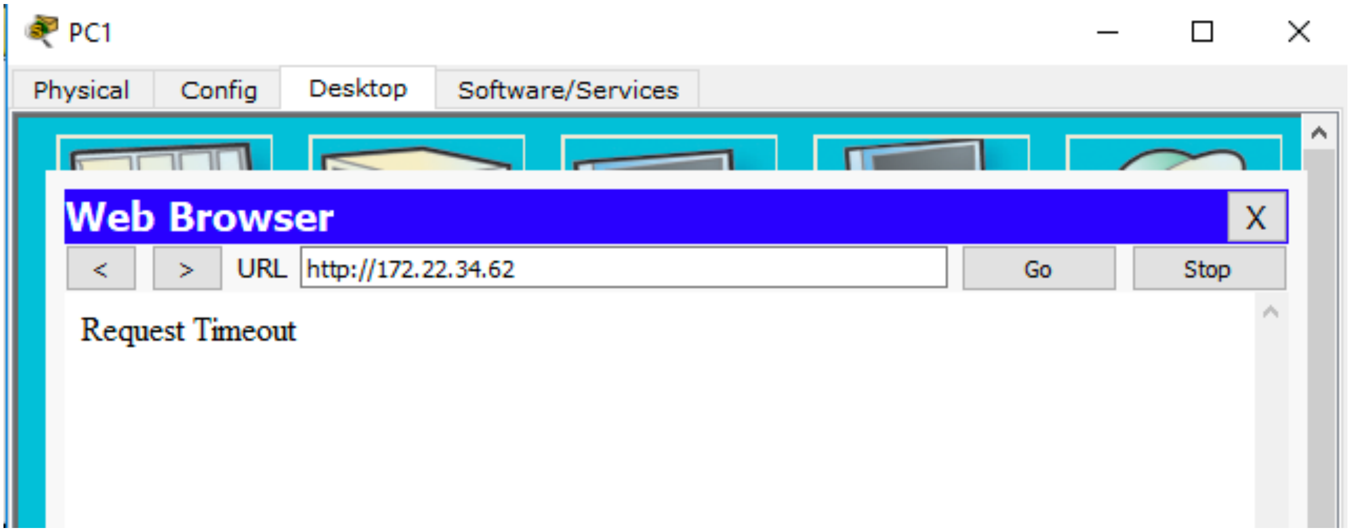
```
ftp> quit
```

d. Ping from **PC1** to **PC2**. The destination host should be unreachable (as shown below), because the traffic was not explicitly permitted.



```
PC>ping 172.22.34.98

Pinging 172.22.34.98 with 32 bytes of data:

Reply from 172.22.34.65: Destination host unreachable.
Reply from 172.22.34.65: Destination host unreachable.
Reply from 172.22.34.65: Destination host unreachable.
Reply from 172.22.34.65: Destination host unreachable.

Ping statistics for 172.22.34.98:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

PC>
PC>
```

e. Try connect from the browser in PC1 to the webserver (from **PC1** to **server)**. The server will be unreachable. Again, this is the case as we didn't permit HTTP traffic to the server.



URL http://172.22.34.62

Request Timeout

Part 2: Configure, Apply and Verify an Extended Named ACL

Step 1: Configure an ACL to permit HTTP access and ICMP to Server from PC2's LAN.

a. Named ACLs start with the **ip** keyword. From global configuration mode of **R1**, enter the following command, followed by a question mark.

```
R1(config)# ip access-list ?
  extended  Extended Access List
  standard  Standard Access List
```

b. You can configure named standard and extended ACLs. This access list filters both source and destination IP addresses; therefore, it must be extended. Enter **HTTP_ONLY** as the name. (For Packet Tracer scoring, the name is **case-sensitive.)**

```
R1(config)# ip access-list extended HTTP_ONLY
```

c. The prompt changes. You are now in extended named ACL configuration mode. All devices on the **PC2** LAN need TCP access. Enter the network address, followed by a question mark. You will now need to calculate a wildcard mask for the network 172.22.34.96/28.

```
R1(config-ext-nacl)# permit tcp 172.22.34.96 ?
  A.B.C.D  Source wildcard bits
```

d. An alternative way to calculate a wildcard is to subtract the subnet mask from 255.255.255.255. **Calculate this wildcard mask and insert it into the command.**

```
  255.255.255.255
- 255.255.255.240
-----------------
=   ?.  ?.  ?. ?
```

e. Finish the statement by specifying the server address as you did in Part 1 and filtering **www** traffic.

```
R1(config-ext-nacl)# permit tcp 172.22.34.96 [Wildcard mask here] host 172.22.34.62 eq
www
```

f. Create a second access list statement to **permit ICMP (ping, etc.) traffic** from **PC2 LAN** to **Server**. Note: The wildcard mask again will be for the whole subnet of 172.22.34.96/28. you will need to insert all the information in [] below.

```
R1(config-ext-nacl)# permit [protocol here] 172.22.34.96 [Wildcard mask here] host
[server address here]
```

g. All other traffic is denied, by default. Exit out of extended named ACL configuration mode.

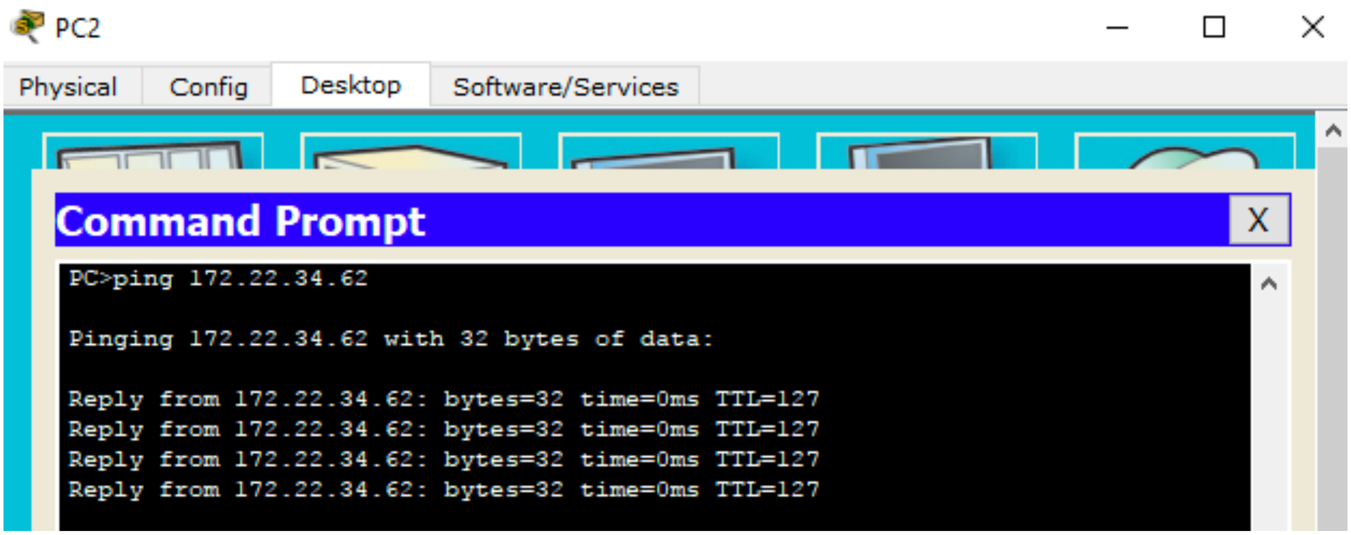
Step 2: Apply the ACL on the correct interface to filter traffic.

Similar to what you did in part 1, you will need to apply the ACL. From **R1**'s perspective, the traffic that access list **HTTP_ONLY** applies to is **inbound** from the network connected to **Gigabit Ethernet 0/1** interface. Apply this configuration. Enter the interface configuration mode and apply the ACL.

```
R1(config)# interface gigabitEthernet [interface port number]
R1(config-if)# ip access-group HTTP_ONLY [direction]
```

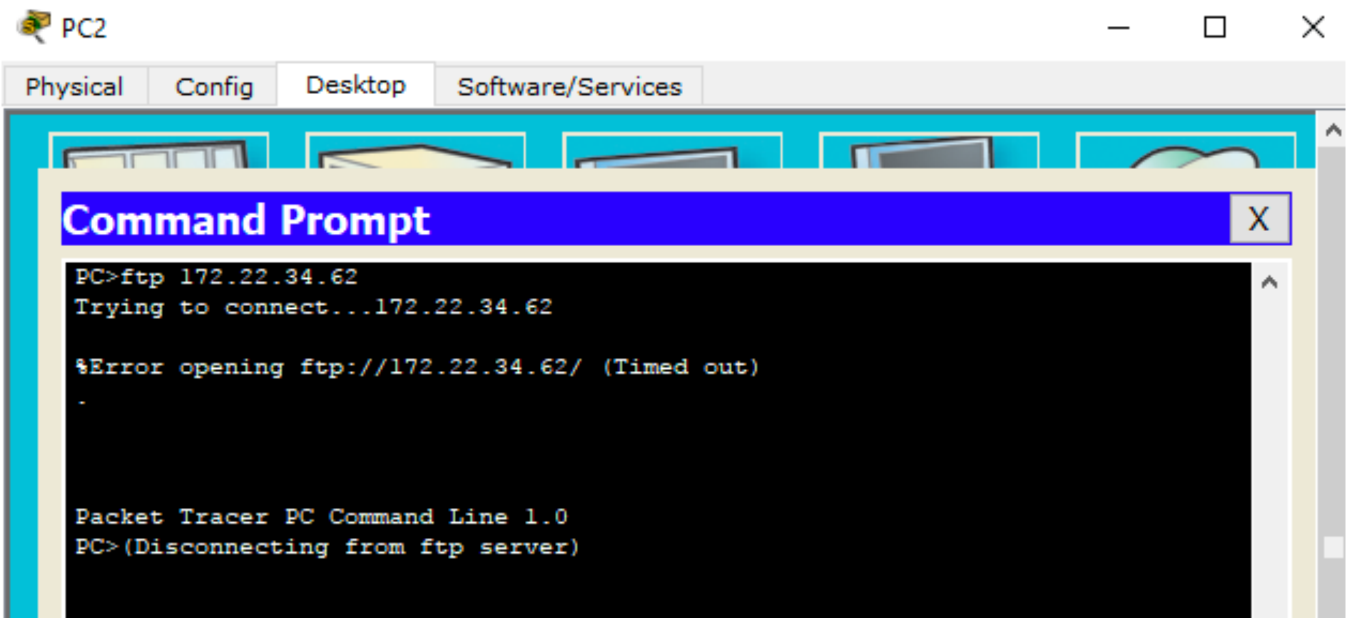
Step 3: Verify the ACL implementation.

a. Ping from **PC2** to **Server**. If the pings unsuccessful, verify the IP addresses before continuing.



```
PC>ping 172.22.34.62

Pinging 172.22.34.62 with 32 bytes of data:

Reply from 172.22.34.62: bytes=32 time=0ms TTL=127
Reply from 172.22.34.62: bytes=32 time=0ms TTL=127
Reply from 172.22.34.62: bytes=32 time=0ms TTL=127
Reply from 172.22.34.62: bytes=32 time=0ms TTL=127
```

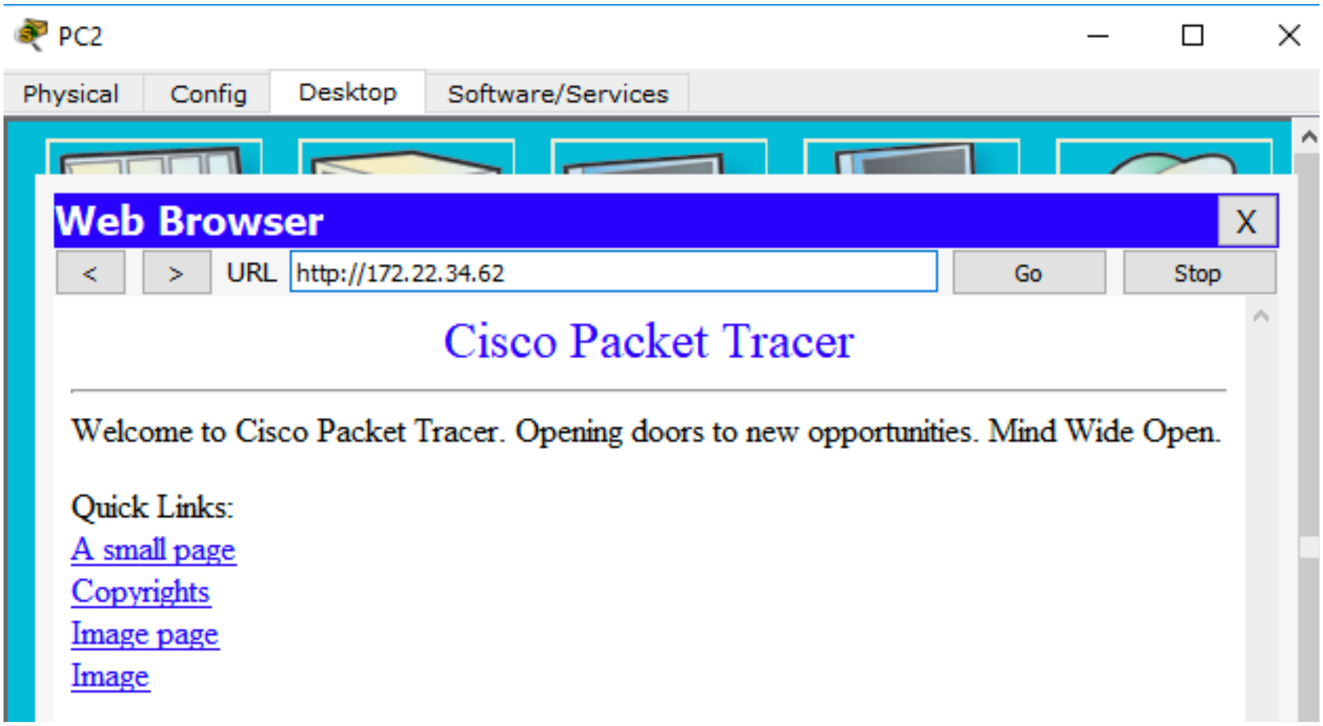
b. FTP from **PC2** to **Server**. The connection should fail (as shown below).



```
PC>ftp 172.22.34.62
Trying to connect...172.22.34.62

%Error opening ftp://172.22.34.62/ (Timed out)
.



Packet Tracer PC Command Line 1.0
PC>(Disconnecting from ftp server)
```

c. Open the web browser on **PC2** and enter the IP address of **Server** as the URL. The connection should be successful.

Task 2 - Configuring Extended ACLs



URL http://172.22.34.62

Cisco Packet Tracer

Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open.

Quick Links:
A small page
Copyrights
Image page
Image