Data Communication and Computer Networks Lecture

Standard ACL,
Extended ACL and Named ACL

Introduction:

In the context of networking and Cisco Packet Tracer, ACL stands for Access Control List. It is a set of rules that determine what traffic is allowed or denied to pass through a network device, such as a router or a switch, based on the defined criteria. ACLs are used for network security purposes to filter and control the flow of network traffic.

There are mainly two types of access control lists: standard access lists and extended access lists. These types can be further classified into two subtypes: numbered and named. A standard access list can be either a numbered standard list or a named standard access list. Similarly, you can have a numbered extended access list or a named extended list.

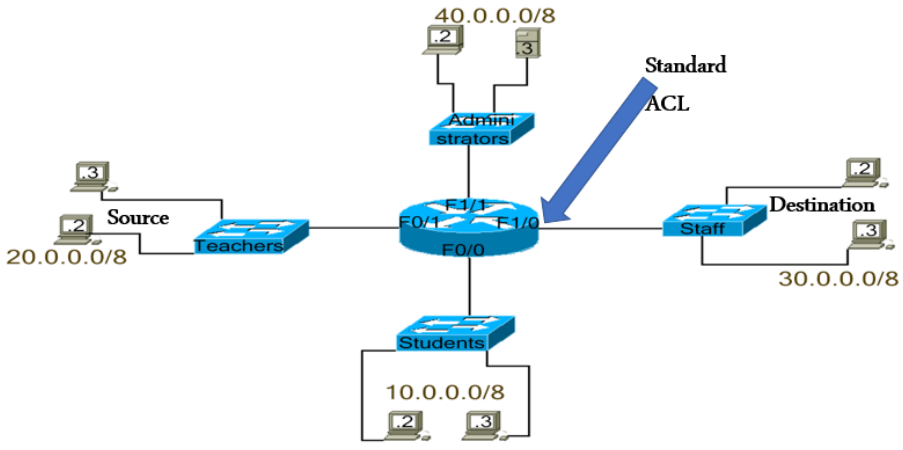
Let us discuss the properties, characteristics, and functions of these types and understand how each type differs from the others.

Standard access lists

Standard access lists are easy to configure. But they support limited options in entries. In a standard access list entry, you can use only the source address to define the criteria. Apart from the source address, you can't use any other option.

Standard access lists work on an 'all or none' formula. They will either allow or block all traffic from the source host. You cannot allow or deny only certain types of traffic from the source host.

Since standard access lists work with all traffic originating from a host, they are applied closer to the destination.



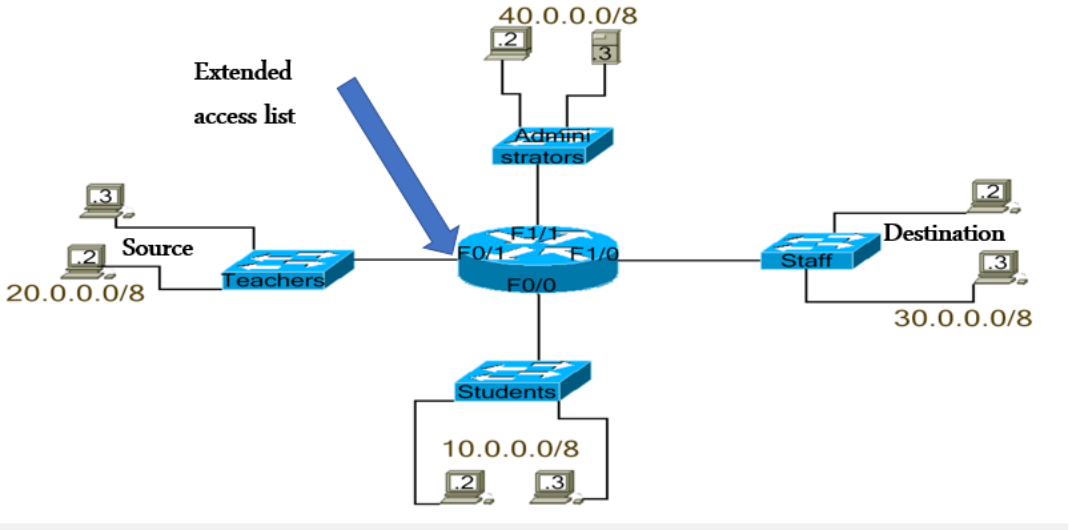
Extended access lists

Extended access lists are complex. But they support many options in entries. In an extended access list entry, you can use a source address, a destination address, protocol, traffic type, application, and port to define the criteria.

Extended access lists allow you to target a specific type of traffic. You can allow a certain type of traffic while blocking the remaining traffic, or you can block a specific type of traffic while allowing the remaining traffic.

Since extended access lists work with a specific type of traffic, they are applied closer to the source.

Extended access list



Numbered and named ACLs

Routers support multiple ACLs. You can create as many ACLs as you want. To differentiate between ACLs, routers use a unique number and name for each ACL. You may consider these numbers or names as identification numbers or names.

When creating an ACL, you must specify an identification number or name for the ACL. Since the router uses this number to identify the ACL, you cannot choose a random number for the ACL. You have to choose a number from a pre-defined range.

Routers reserve the following number ranges for standard access lists and extended access lists.

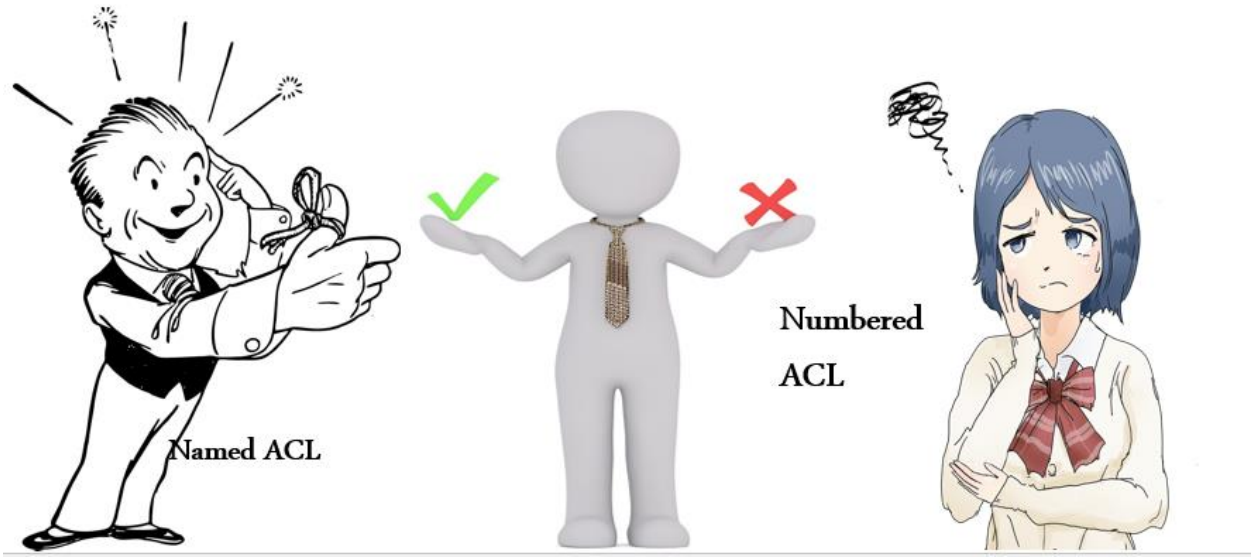
```
Standard access lists 1 - 99 and 1300 - 1999
Extended access lists 100 - 199 and 2000 - 2699
```

To create a standard access list, you can use any number from the range 1 - 99 and 1300 - 1999. For example, you can use the number 10 or 1400, but you cannot use the number 150 or 2100.

Similarly, to create an extended list, you can use any number from the range 100 - 199 and 2000 - 2699. For example, you can use the number 120 or 2450, but you cannot use the number 50 or 1500.

Numbers are a bit difficult to remember. They also do not provide any descriptive meaning. If you have multiple ACLs, it becomes very difficult to remember which ACL is doing what. To make ACLs management easier, routers also support names for ACLs. It means you can also use descriptive names for ACLs instead of pre-defined numbers.

No matter whether you use a name or a number for the ACL, the ACL functions the same way. As far as functionality is concerned, named ACLs and numbered ACLs are the same. The main advantage of a named ACL over a numbered ACL is that a named ACL is easier to manage and remember than a numbered ACL.



Let's take an example. You check the configuration of a router and find the following ACLs.

Interface	ACL	Direction
F0/0	25	Inbound
S0/0/0	145	Outbound
S0/0/1	39	Inbound

To figure out what these ACLs are doing, you have to check the entries of each ACL. Now, suppose, you read the configuration of another router and find the following ACLs.

Interface	ACL	Direction
F0/0	BlockingStudents	Inbound
S0/0/0	AllowingAdmin	Outbound
S0/0/1	BlockingExternalUsers	Inbound

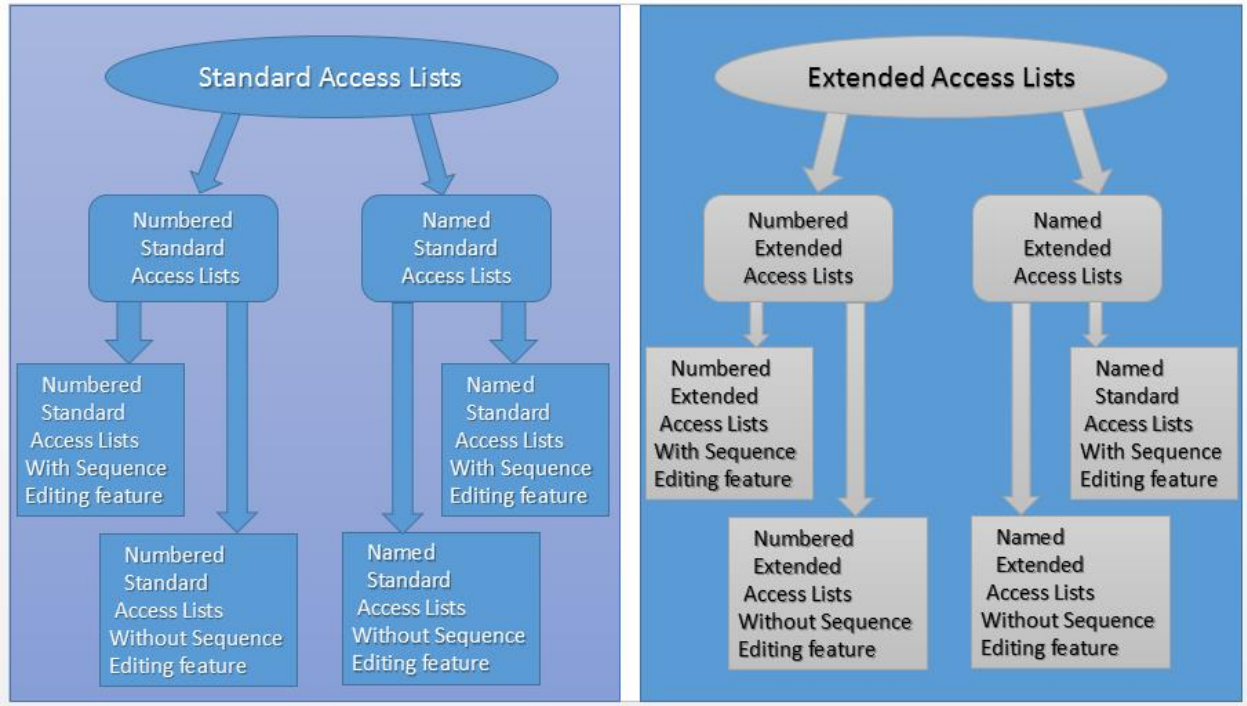
By looking at these ACLs you can get an idea of what each ACL is doing. For example, by looking at the name BlockingStudents, you can guess that this ACL would be blocking traffic from the Students segment.

By using a descriptive name (such as block-external-users), a network administrator can easily determine the purpose of the ACL. This feature is especially helpful in large networks, where a router may have multiple ACLs with hundreds of statements.

Advanced sequence editing ACLs

Advanced sequence editing is a new feature. Before this feature, editing or updating ACL entries was not possible. To edit an ACL entry, you had to recreate the entire ACL. This feature allows an administrator to change, update, or delete a single entry from an ACL. This feature was added later to Cisco IOS. All new IOS versions include this feature. If the IOS includes this feature, you can use this feature to edit both types of ACL.

The following image shows all types of Cisco access lists.



How Access Lists work on Cisco routers

when routers receive IP packets on their interfaces, they check the destination address of each packet and forward that packet from the interface that is directly connected to the destination address or the path

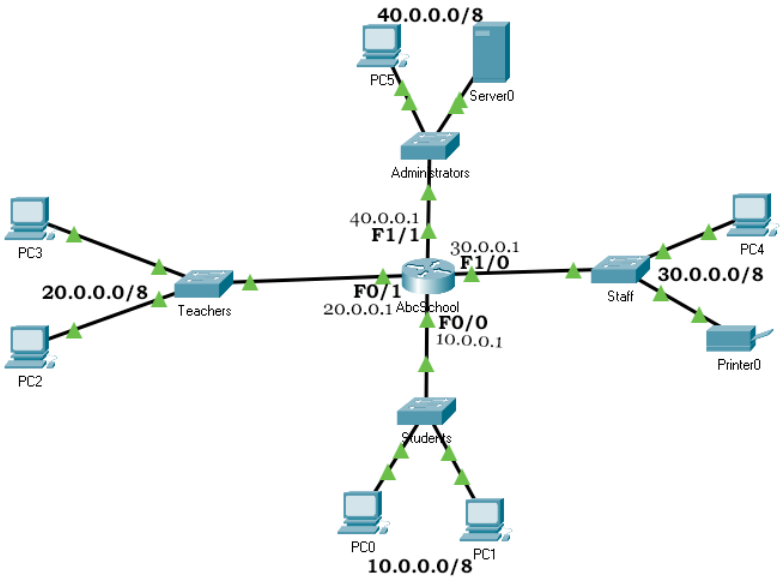
leading to the destination address. If none of the router's interfaces are connected to the destination address, the router discards the packet.

An access list is a set of additional commands or instructions that you can instruct a router to perform before forwarding IP packets. In the access list, each command or instruction is written on a separate line. Each line of the access list is treated as a separate entry.

An access list can contain many entries. Each entry must include a criterion and an action. A criterion defines the condition that triggers the action. An entry may include multiple criteria or actions.

Let's take an example to understand how access lists work.

The following image shows a sample school network.



In this network, four LAN segments are connected through a router. These segments are Students, Teachers, Staff, and Administrators. These segments respectively belong to students, teachers, office staff, and management team.

The following table lists the IP configurations of all segments.

LAN	Network Address	Default gateway	Gateway interface
Students	10.0.0.0/8	10.0.0.1	Router'F0/0
Teachers	20.0.0.0/8	20.0.0.1	Router'F0/1
Staff	30.0.0.0/8	30.0.0.1	Router'F1/0
Administrators	40.0.0.0/8	40.0.0.1	Router'F1/1

As far as connectivity is concerned, this network is fine. All LAN segments can access each other without any issues. The main issue of this network is security. This network has no security policy. Anyone can

access any resource of the network. A student can access the teacher's computer. A teacher can access the principal's computer. This free flow of access makes this network useless. This network will be useful only if it allows only authorized users to access permitted resources.

To block unauthorized access, Cisco routers have a built-in feature. This feature is known as access-lists. An access list allows the administrator to define what is allowed and what is blocked.

Once the criteria for allowed packets are defined, the router will only allow packets that meet the defined criteria. Access lists are used to define criteria for allowed packets. Access lists use lines to separate entries. Each line in the access list represents an entry. Each entry contains two things a condition and an action. When processing the entry, the router matches the condition, if the condition is matched, the router executes the action. A condition may include a single criterion or multiple criteria.

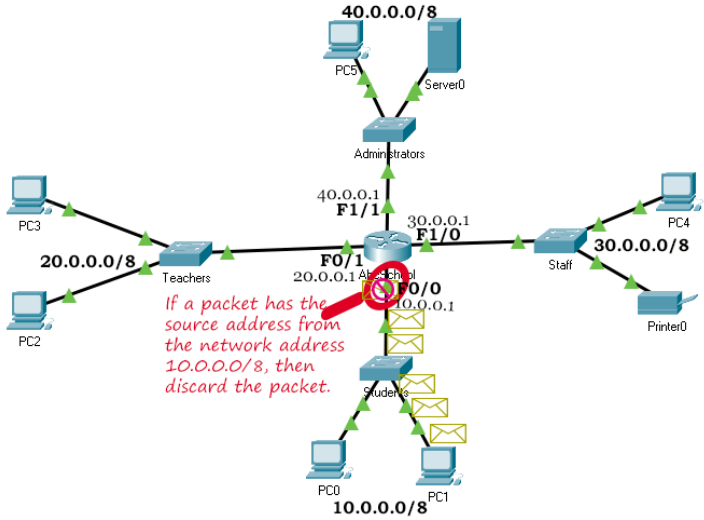
When creating an access list entry, you should keep three important factors in mind. These factors are location, direction, and order. We have already discussed these factors in the previous part of this lecture. In this part, we will take an example to understand how an access list works and how these factors can affect an access list.

Location

In our example network, all LAN segments can access all LAN segments. To block students from accessing resources available outside the Students segment, the administrator created an access list and applied it to the F0/0 interface of the router. The ACL has the following entry.

If a packet has the source address from the network address 10.0.0.0/8, then discard the packet.

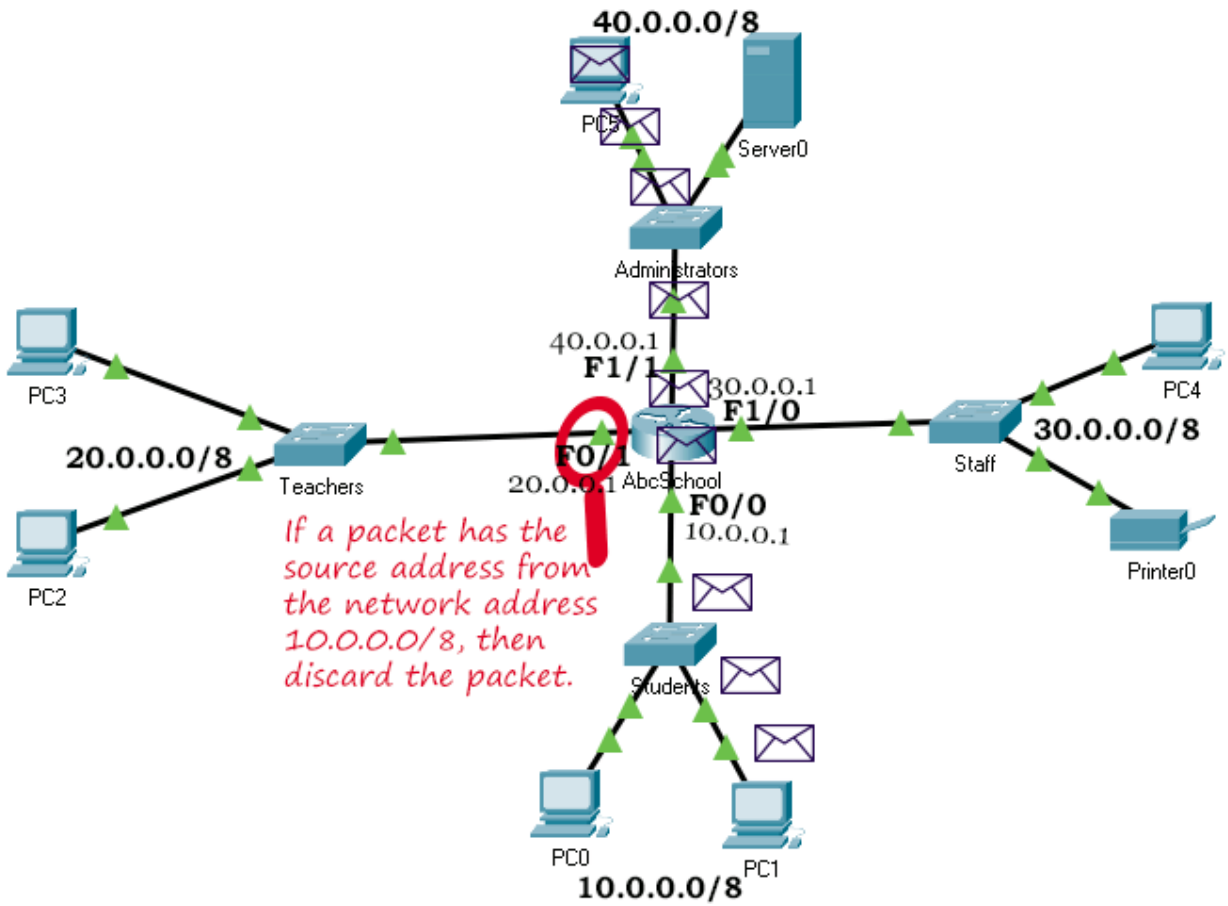
The following image shows how this ACL is applied.



Since all packets generated from the Students segment have source addresses from the 10.0.0.0/8 network, they will be blocked as soon as they enter the F0/0 interface. After this ACL, users from the Students segment will not be able to access outside resources.

To understand how the location affects the ACL, let's suppose the administrator applied the above ACL to the F0/1 interface.

The following image shows this change.



Now, this ACL is useless. This ACL instructs the router to block a packet if it arrives from the 10.0.0.0/8 network. A packet from the 10.0.0.0/8 will never enter from the F0/1 interface. The F0/1 interface is the default gateway of the Teachers segment. Since the network address of the Teachers segment is 20.0.0.0/8, all packets entering F0/1 will have a source address from the network 20.0.0.0/8.

This example shows how the correct location of the ACL is important. An ACL must be implemented on the interface that interacts with targeted traffic.

Direction

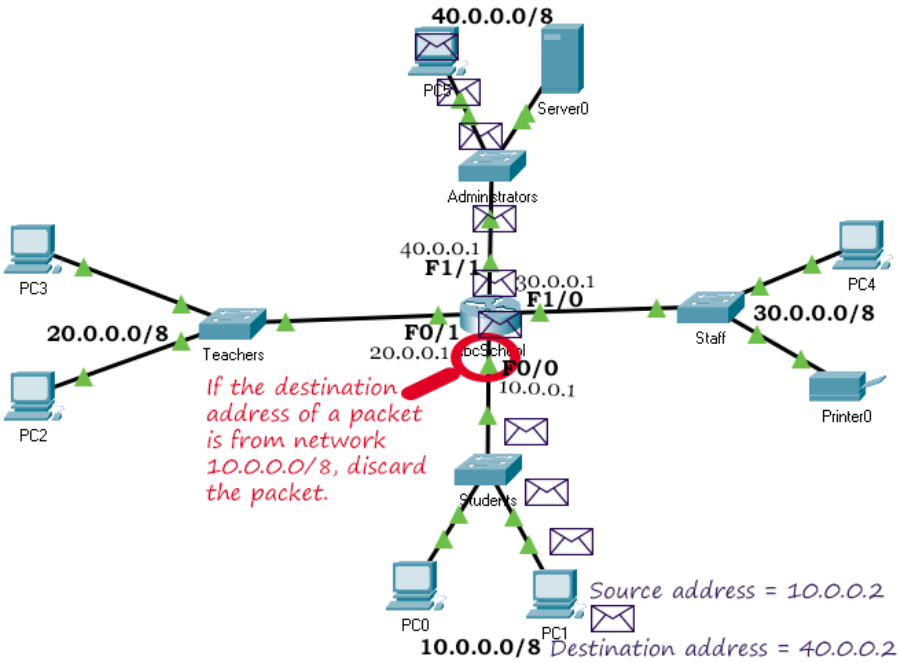
Now suppose, instead of using the source address in the ACL entry, the administrator mistakenly used the destination address. The modified ACL is given below.

If the destination address of a packet is from the network 10.0.0.0/8, discard the packet.

The administrator applied this ACL to the F0/0 interface of the router.

Will this ACL work?

The following image shows the new ACL.



This ACL will not work. This ACL instructs the router to block the packets that are going to the network 10.0.0.0/8, not to the packets that are coming from the network 10.0.0.0/8. If you apply this ACL to the F0/0 interface, the Students segment will be able to access all three segments but they will not be able to access the Students segment.

Order

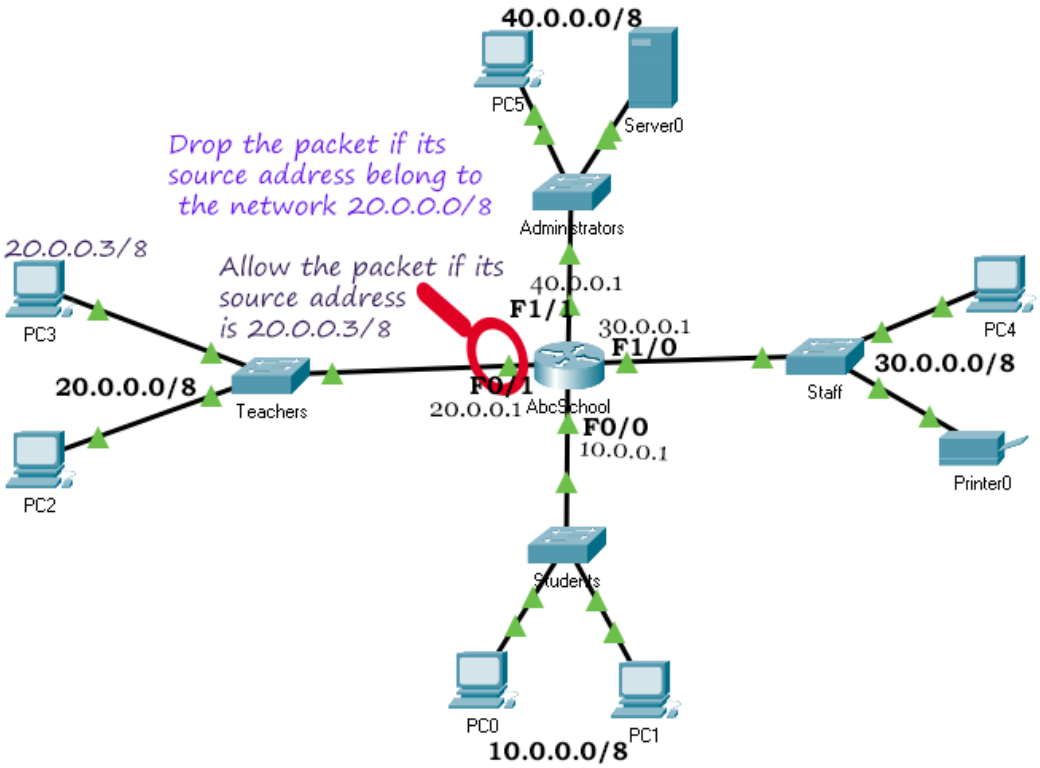
As mentioned earlier, for each packet, the router checks ACL entries from top to bottom until a match is found. Once a match is found, it does not check the remaining entries for that packet. Let's understand this factor through the example.

The administrator wants to allow a user from the Teachers segment to access the server available in the Administrators segment. The IP address of the allowed user is 20.0.0.3/8. Apart from the allowed user, all remaining users must not be able to access the Administrators segment. For this, the administrator created the following ACL and applied it to the F0/1 interface of the router.

Drop the packet if its source address belongs to the network 20.0.0.0/8

Allow the packet if its source address is 20.0.0.3/8

The following image shows this ACL.



Will this ACL work?

No, this ACL will block all outgoing traffic from the Teachers segment. When a packet originated from the host 20.0.0.3/8 reaches the router, the router checks the entries of the applied ACL until a match is found.

The first line of the ACL says "drop the packet if its source address belongs to the network 20.0.0.3/8". Since the IP address 20.0.0.3/8 belongs to the network 20.0.0.0/8, the statement becomes true. The router executes the action that is associated with this statement. Since the action of this statement is the drop, the router drops the packet.

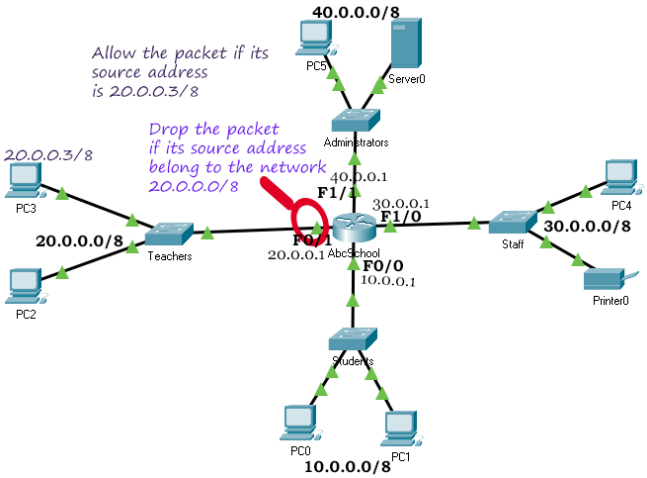
Even the second line of the ACL allows the host 20.0.0.3/8, but it will never be read and executed by the router.

The correct order to allow the host 20.0.0.3/8 will be the following.

Allow the packet if its source address is 20.0.0.3/8

Drop the packet if its source address belongs to the network 20.0.0.0/8

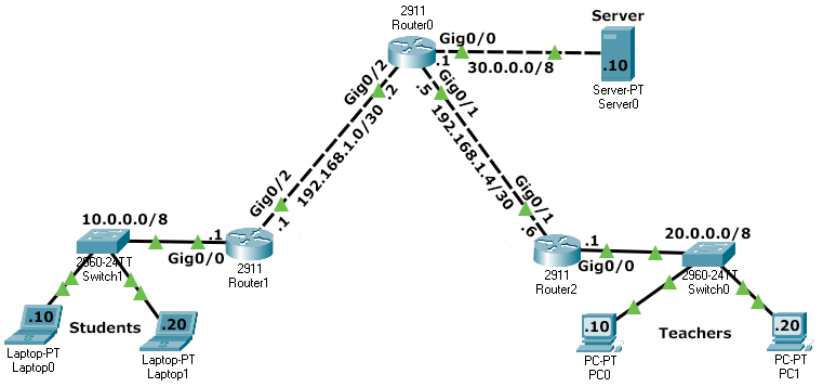
The following image shows the above ACL.



Now, this ACL will allow all packets that are originated from the host 20.0.0.3/8 but it will block all packets that are originated from other hosts of the network 20.0.0.0/8.

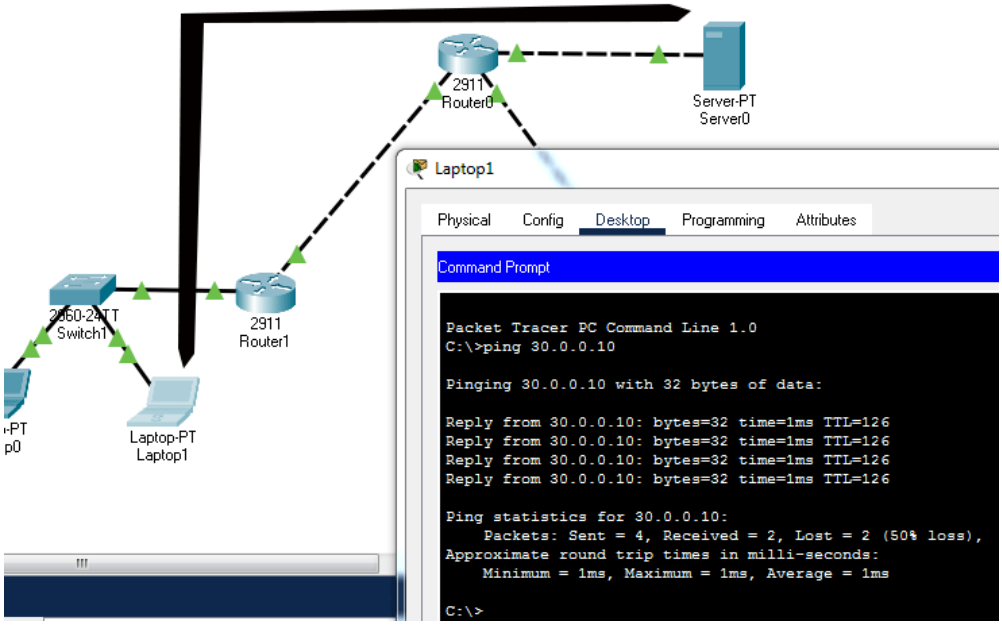
Configure Standard Access Control

Create a packet tracer lab as shown in the following image.



Configure IP addresses as shown in the above image and enable RIPv2 protocol for routing and test connectivity between sections. To test connectivity between sections, you can use the **ping** command.

The following image shows how to use the **ping** command to test connectivity between **Laptop1** and **Server0**.



If all end devices can access each other, the lab is ready for practice. If you have a connectivity-related issue or can't replicate this lab.

Objectives/requirements

Create and implement a standard access list that blocks the Students section from accessing the Server section.

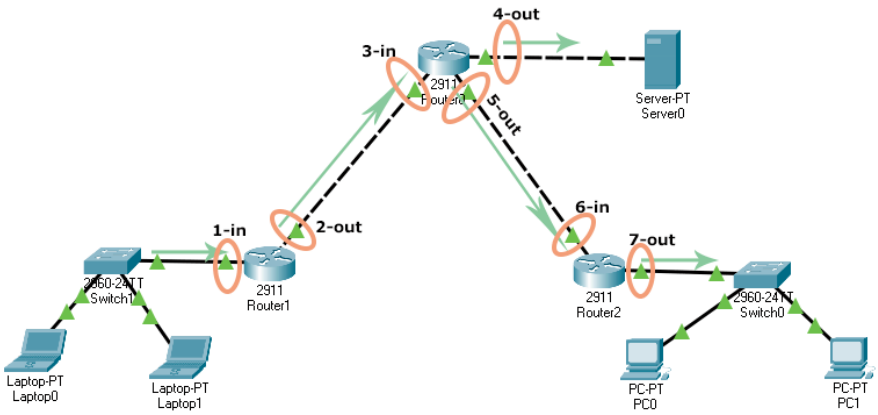
Understanding requirements

The Students section uses IP subnet 10.0.0.0/8. All packets originating from this section have an IP address from this subnet. If we create a standard ACL with a deny statement for this subnet, all packets having an IP address from this subnet in their source address will be dropped.

Selecting location and direction for the ACL

A router's interface uses the ACL to filter traffic passing through it. An incorrectly implemented ACL can block entire traffic passing through it. Before creating and implementing an ACL, we have to select the correct interface and the correct direction for the ACL.

In our network, we have seven locations where we can implement the ACL. The following image shows these locations and the direction in which they can be used to filter traffic.



The following table lists the above locations and the effect of the ACL on each location.

Location	Interface	Direction	Effect
1	Router1's Gig0/0	In	The Students section will not be able to access the Server and the Teachers section.
2	Router1's Gig0/2	Out	The Students section will not be able to access the Server and Teachers section.
3	Router0's Gig0/2	In	The Students section will not be able to access the Server and Teachers section.
4	**Router0's Gig0/0**	**Out**	**The Students section will not be able to access the Server section but it will be able to access the Teachers section.**
5	Router0's Gig0/1	Out	The Students section will not be able to access the Teachers section but it will be able to access the Server section.
6	Router1's Gig0/1	In	The Students section will not be able to access the Teachers section but it will be able to access the Server section.
7	Router1's Gig0/0	Out	The Students section will not be able to access the Teachers section but it will be able to access the Server section.

As you can see in the above table, the correct location for our ACL is Router0's Gig0/0 and the correct direction is the out.

Standard ACL configuration commands

We have two commands to create a standard access list. These commands are **'access-list'** and **'ip access-list'**. The **'ip access-list'** command has an advantage over the **'access-list'** command. It allows us to update or modify statements. We have already learned how to use the **'access-list'** command to create a standard access list in the previous part of this lecture. In this part, let's use the **'ip access-list'** command.

The **'ip access-list'** is a global configuration mode command. To create a standard access list, it uses the following syntax.

```
Router(config)# ip access-list standard ACL_#
```
In the above syntax, the **ACL_#** is the name or number of the standard ACL. When you hit the enter key after entering this command, the command prompt changes and you enter standard ACL configuration mode.

```
Router(config-std-acl)#
```
In standard ACL configuration mode, you can use the following syntax to create statements.

```
Router(config)# ip access-list standard ACL_name
Router(config-std-acl)# permit|deny source_IP_address
[wildcard_mask]
```
An ACL does nothing until it is applied to an interface. To apply a standard ACL to an interface, enter the interface configuration mode of the interface and use the following command.

```
Router(config)# interface type [slot_#]port_#
Router(config-if)# ip access-group ACL_# in|out
```
Once an ACL is activated on an interface, the interface processes all packets through it.

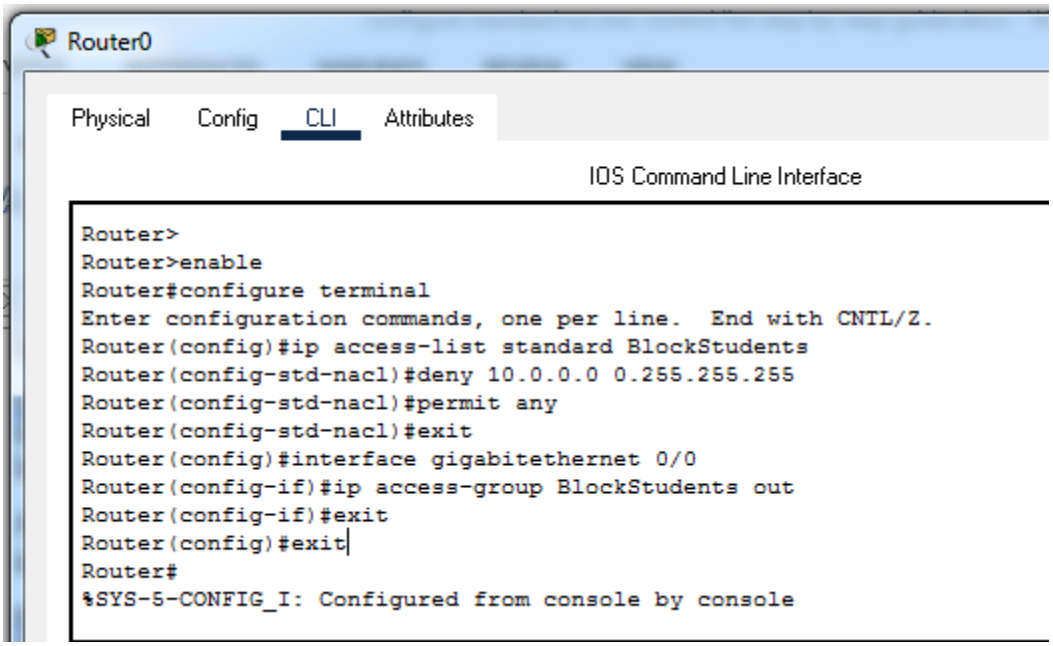
Creating a standard ACL

Access the command prompt of Router0 and run the following commands.

```
Router>
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ip access-list standard BlockStudents
Router(config-std-nacl)#deny 10.0.0.0 0.255.255.255
Router(config-std-nacl)#permit any
Router(config-std-nacl)#exit
Router(config)#interface gigabitethernet 0/0
Router(config-if)#ip access-group BlockStudents out
Router(config-if)#exit
Router(config)#exit
Router#
```

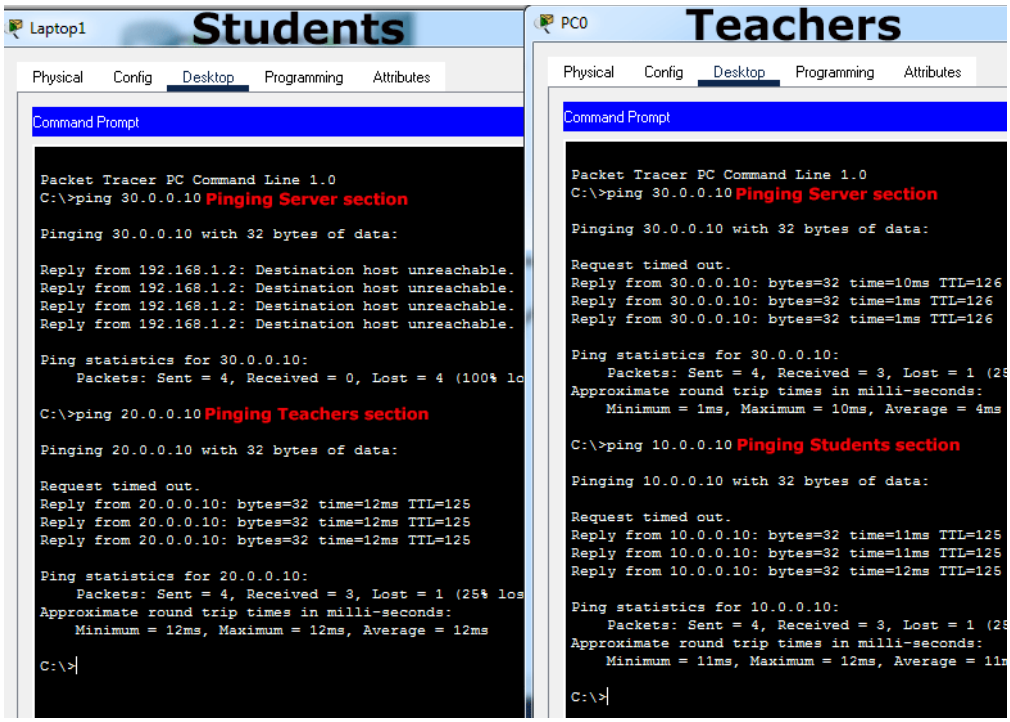
Let's discuss the above commands. We used the first two commands to enter global configuration mode. The next command creates a standard ACL named BlockStudents. In ACL configuration mode, we added two statements. The first statement denies all traffic from the 10.0.0.0/8 subnet. The second statement allows all other traffic. We used the next commands to exit ACL configuration mode and enter interface configuration mode. The next command applies the BlockStudents ACL in the out direction. The last two commands exit interface configuration mode and global configuration mode, respectively.

The following image shows how to run the above commands on the command prompt of the router.



Verifying

To verify the ACL, we can test connectivity between sections. The Students section should not be able to access the Server section but it should be able to access the Teachers section. The Teachers section should be able to access both the Server and the Students section. You can use the ping command to test connectivity. The following image shows this testing.



Modifying /updating a standard ACL statement

To modify or update a standard ACL statement, use the following steps.

- Use the '**show access-lists**' command to view the sequence number of the statement.
- Enter standard ACL configuration mode
- Delete the existing statement with the '**no [*sequence number*]**' command
- Insert the modified, updated, or the new statement with the sequence number of the old statement

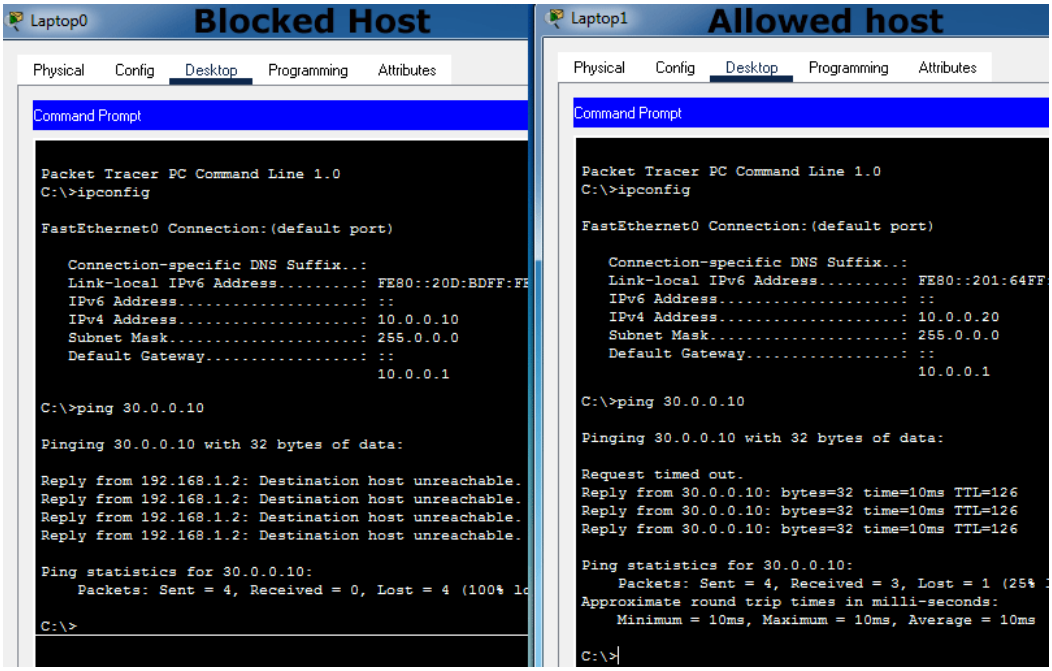
Let's take an example. Suppose, instead of blocking the entire subnet we only want to block a single host (10.0.0.10/8) from the Students section. For this, access the CLI prompt of Router0 and run the following commands.

```
Router>
Router#show access-lists
Standard IP access list BlockStudents
10 deny 10.0.0.0 0.255.255.255
20 permit any
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ip access-list standard BlockStudents
Router(config-std-nacl)#no 10
Router(config-std-nacl)#10 deny 10.0.0.10 0.0.0.0
Router(config-std-nacl)#exit
Router(config)#exit
Router#
Router#show access-lists
Standard IP access list BlockStudents
10 deny host 10.0.0.10
20 permit any
Router#
```

Let's understand the above commands.

First, we checked the sequence number of the statement that we had used to block the entire Students section. As we can in the above output, the sequence number of the statement is 10. After it, we entered the ACL configuration mode of the ACL. In ACL configuration mode, we deleted the current statement with the 'no sequence_number_of_statement' command. In the end, we inserted the new statement at the place of the existing statement.

Since the ACL is already active on the interface, the interface starts using the new statement as soon as it is added. To verify the change, send ping requests again from the blocked host and the allowed host. The following image shows this testing.



Deleting a standard ACL

To delete a standard ACL, use the following command in global configuration mode.

```
Router(config)no ip access-list standard ACL_#
```

Replace **ACL_#** with the ACL name or number.

The following command deletes the **BlockStudents** ACL.

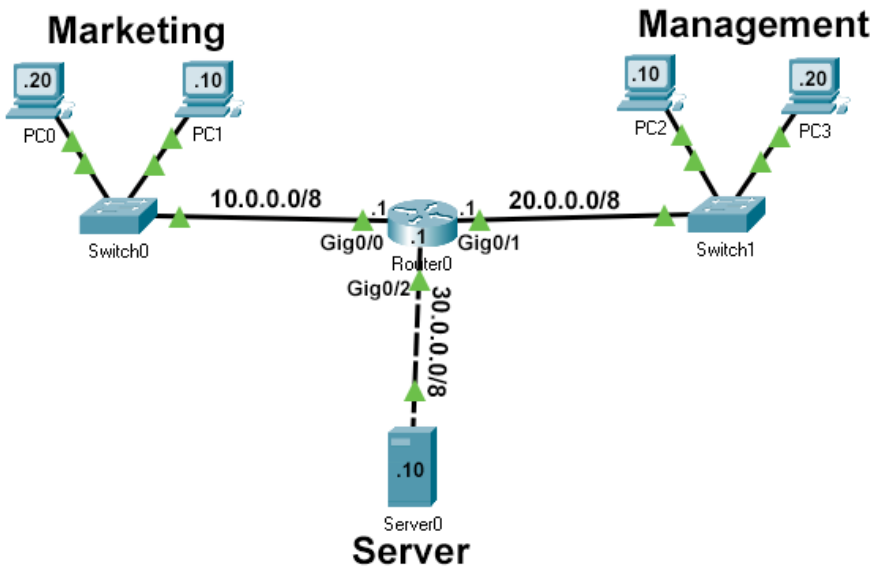
```
Router(config)no ip access-list standard BlockStudents
```

Configure Extended Access Control List

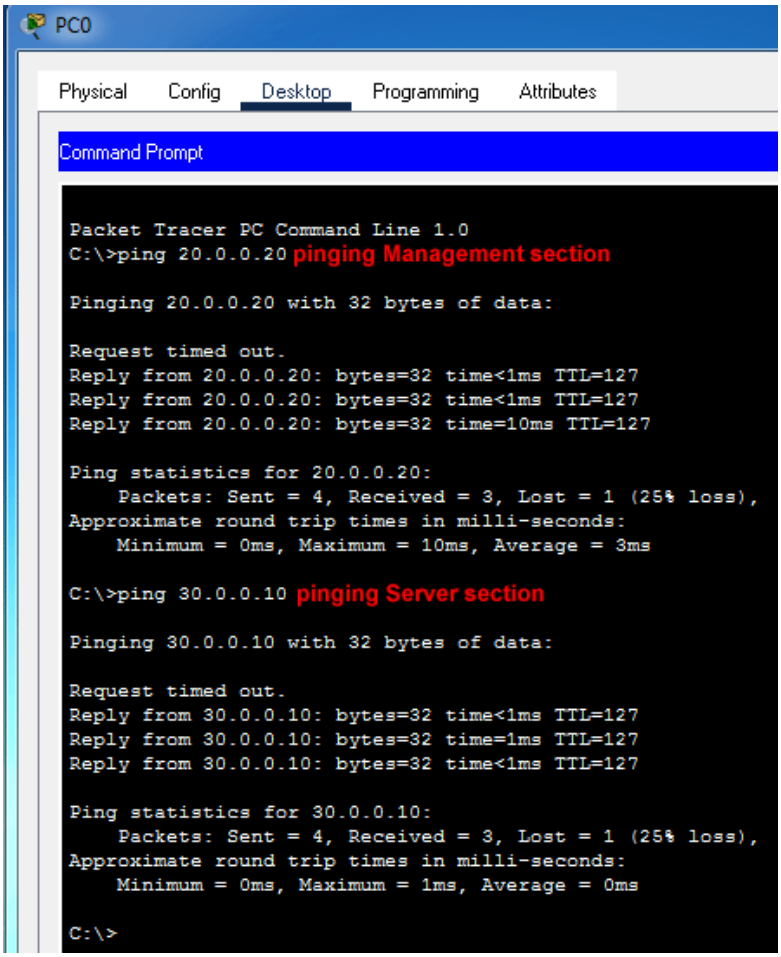
Extended access lists are flexible. They support many options and parameters to define criteria in statements. For example, you can use a source address, a destination address, a layer-3 protocol, and a layer-4 protocol.

In this lecture, we will discuss how to define criteria for layer-4 protocols in extended access lists. In an IP network, two protocols work on layer 4. These protocols are TCP and UDP. We will learn how to create an extended access list for both protocols.

Create a practice lab on Packet Tracer as shown in the following image.

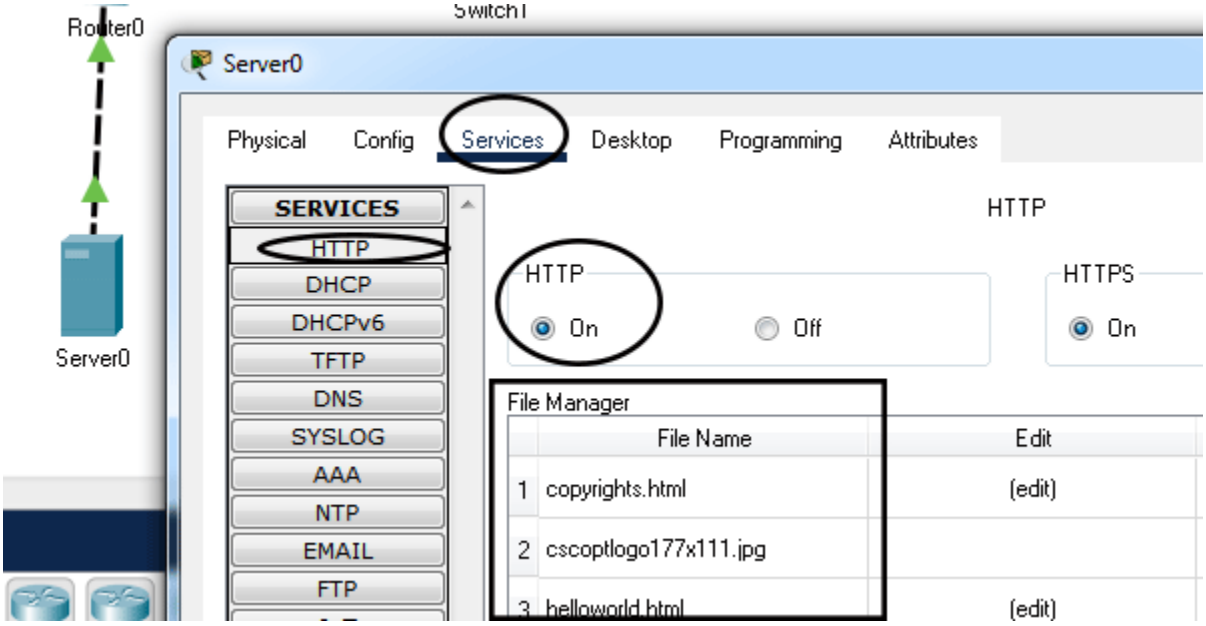


Configure IP addresses as shown in the above image and test connectivity between sections. To test connectivity, you can use the **'ping'** command. The following image shows testing from PC0.

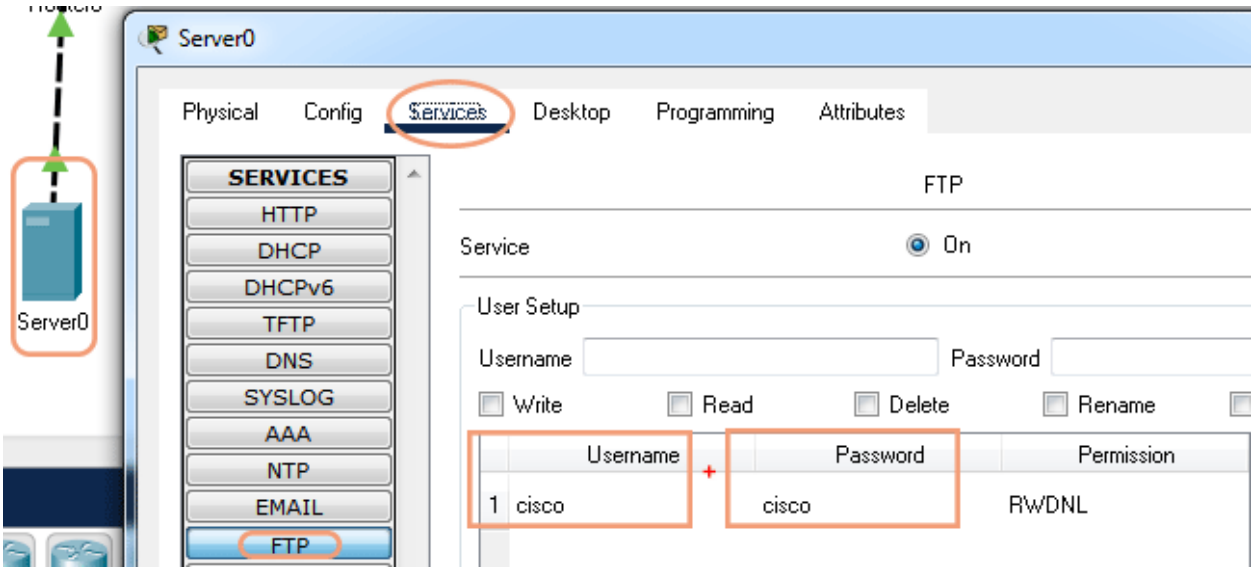


Server0 includes many services. From these services, we will use three services to test layer-4 connectivity. These services are HTTP, FTP, and DNS.

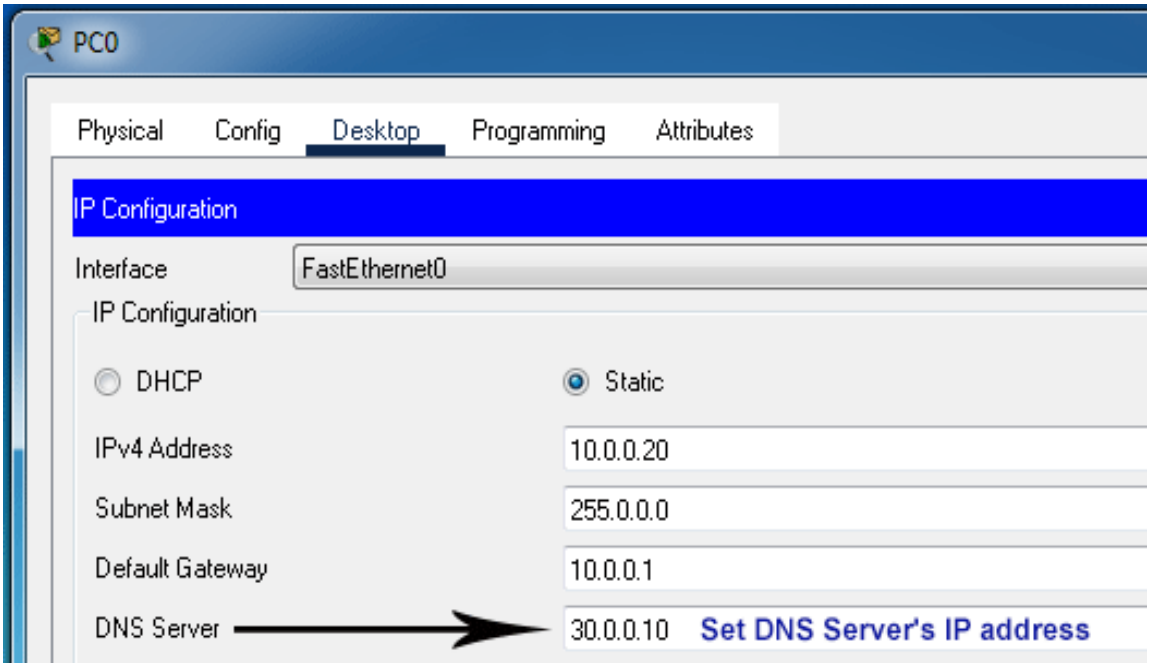
The HTTP service is already enabled. We don't need to make any changes to enable this service.



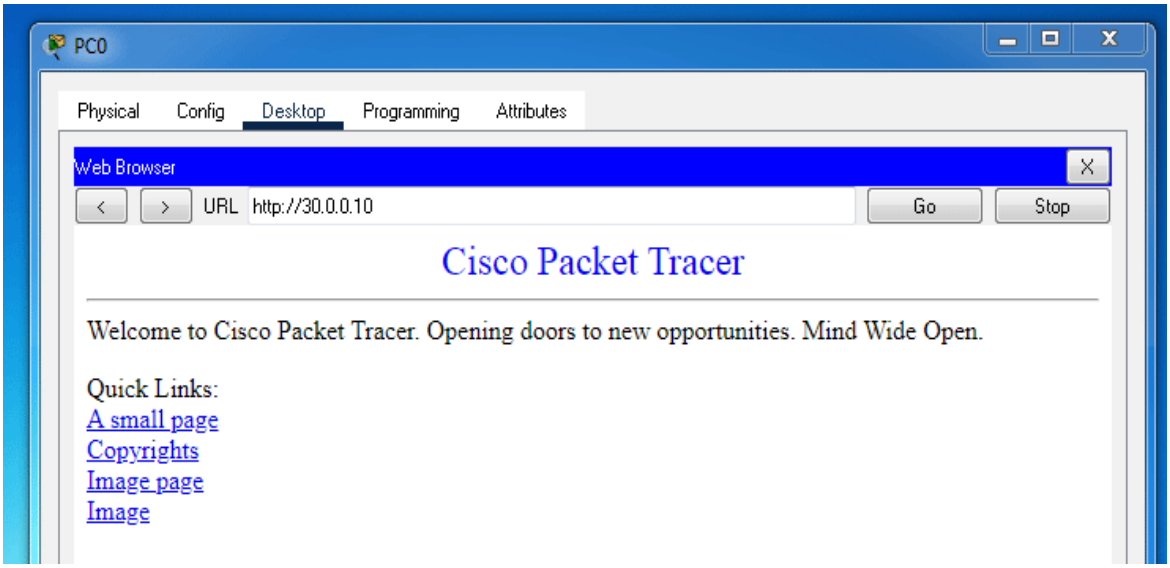
Just like the HTTP service, the FTP service is also enabled by default. The FTP service requires authentication. For testing, a default account is also created. The username and password for this account are **'cisco'** and **'cisco'**, respectively.



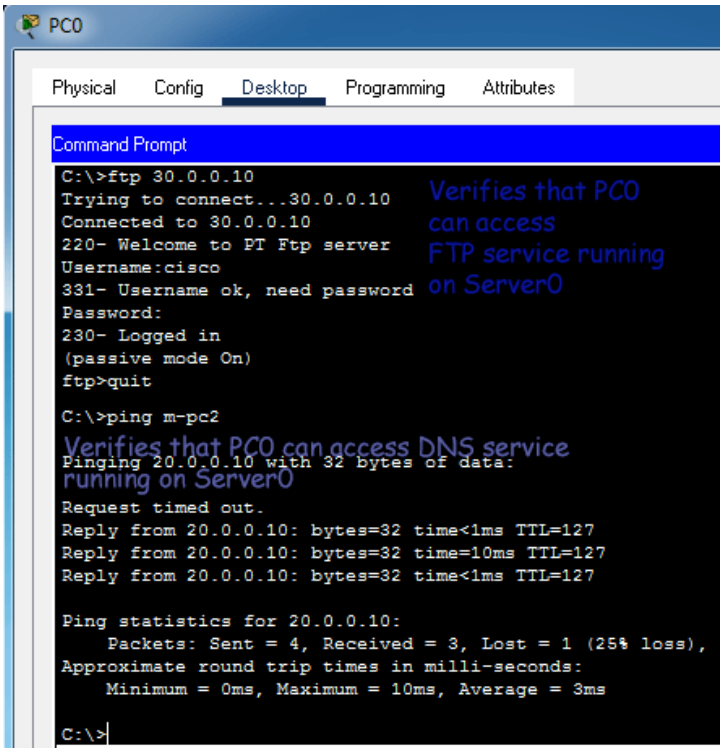
We also have to update the IP configuration on PCs to make them DNS clients. Add the DNS server's IP address to the IP configuration of PCs. The following image shows how to set the DNS server's IP address on PC0.



After updating the DNS server's IP address, verify that PC0 can access all three services. The following image verifies that PC0 can access web service running on Server0.



The following image verifies that PC0 can access FTP and DNS services running Server0.



Requirements

Create an extended access list that allows the Marketing section to access only the web service and DNS service from the Server. The Marketing section should not be allowed to access any other services running on the Server.

Understanding requirements

To fulfill the above requirements, we have to add the following statements to the extended access list.

- A statement that allows access to the web service.
- A statement that allows access to the DNS service.
- A statement that blocks access to all other services.
- A statement that allows access to the Management section.
- A statement that blocks all other traffic.

An extended list is applied near to the source. In our example, we want to filter the traffic that originates from the Marketing section. The Marketing section's traffic enters the network from the Gig0/0 interface of the router. We will implement an extended ACL on this interface with the above statements.

Port numbers/names

To keep each application's data separate from other applications, TCP and UDP assign a unique numeric value to each application. This value is known as the port number. We use the port number of an application to match the traffic of that application.

Some applications also use keywords. If a keyword is available, you can use the keyword in the place of the port number. Since keywords are not available for all applications, it is recommended to use port numbers instead of names.

The following table lists port numbers and names for some most common applications.

Application	Protocol	Port number	Keyword
FTP	TCP	21	ftp
Telnet	TCP	23	telnet
SMTP	TCP	25	smtp
HTTP	TCP	80	www
POP3	TCP	110	pop3
DNS	UDP	53	dns
TFTP	UDP	69	tftp
SNMP	UDP	161	snmp
IP RIP	UDP	520	rip

Creating an extended access list

There are two commands to create an extended access list. These commands are **'access-list'** and **'ip access-list'**. We have already discussed the **'access-list'** command in the previous part of this article. In this part, we will use the **'ip access list'** command to create the extended access list.

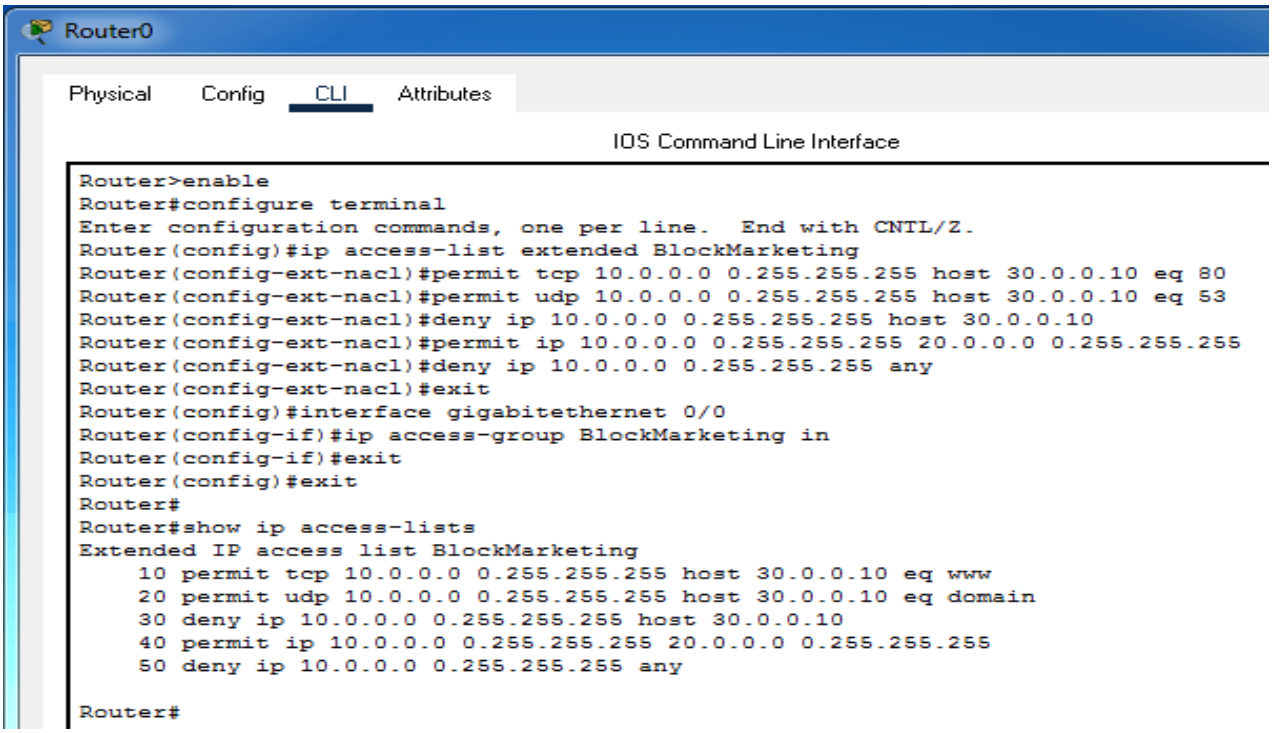
Access the command line interface of the Router and run the following commands.

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ip access-list extended BlockMarketing
Router(config-ext-nacl)#permit tcp 10.0.0.0 0.255.255.255 host
30.0.0.10 eq 80
Router(config-ext-nacl)#permit udp 10.0.0.0 0.255.255.255 host
30.0.0.10 eq 53
Router(config-ext-nacl)#deny ip 10.0.0.0 0.255.255.255 host
30.0.0.10
Router(config-ext-nacl)#permit ip 10.0.0.0 0.255.255.255
20.0.0.0 0.255.255.255
Router(config-ext-nacl)#deny ip 10.0.0.0 0.255.255.255 any
Router(config-ext-nacl)#exit
Router(config)#interface gigabitethernet 0/0
Router(config-if)#ip access-group BlockMarketing in
Router(config-if)#exit
Router(config)#exit
Router#
```

The above commands create an extended access list **BlockMarketing** and apply it to the GigabitEthernet 0/0 interface in the inward direction. The access list contains five statements. The following table lists the meaning of these statements.

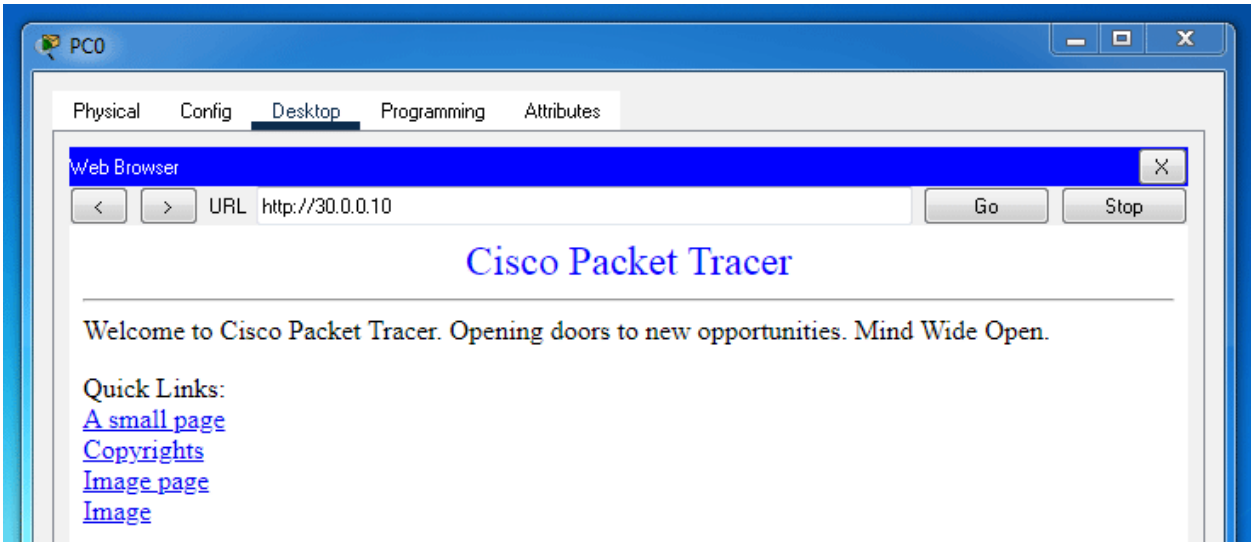
Statements	Description/action
permit tcp 10.0.0.0 0.255.255.255 host 30.0.0.10 eq 80	Allow a packet if its source address is from the network 10.0.0.0/8 and the destination address is 30.0.0.10 and the destination application is HTTP.
permit udp 10.0.0.0 0.255.255.255 host 30.0.0.10 eq 53	Allow a packet if its source address is from the network 10.0.0.0/8 and the destination address is 30.0.0.10 and the destination application is FTP.
deny ip 10.0.0.0 0.255.255.255 host 30.0.0.10	Block a packet if its source address is from the network 10.0.0.0/8 and the destination address is 30.0.0.10.
permit ip 10.0.0.0 0.255.255.255 20.0.0.0 0.255.255.255	Allow a packet if its source address is from the network 10.0.0.0/8 and the destination address is from the network 20.0.0.0/8.
deny ip 10.0.0.0 0.255.255.255 any	Block a packet if its source address is from the network 10.0.0.0/8 and the destination address is from any network.

The following image shows how to execute the above commands on the Router.



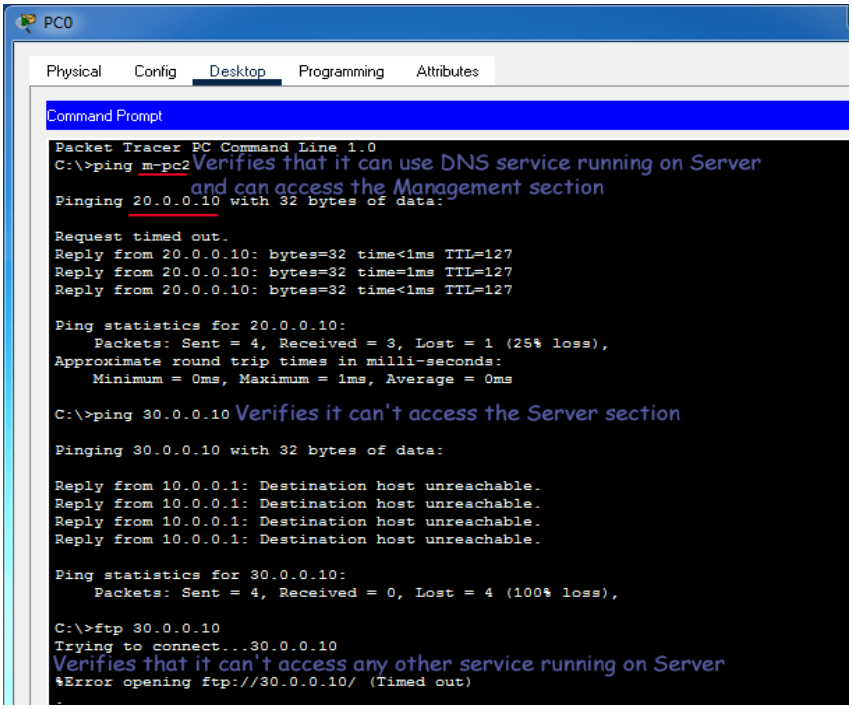
Testing/verifying the extended access list

To verify that the Marketing section can access the webserver running on Server0, you can access a web page from the webserver. The following image shows how to perform this test on PC0.



To verify that the Marketing section can access the DNS service running on Server0 and can access the Management section, you can send ping requests to a PC of the Management section from PC0. To send ping requests, instead of using the IP address of the PC, use the name of the PC. The ping command will use the DNS service running on Server0 to resolve the name to the IP address and then will send ping requests to the IP address. This way, you can verify both requirements with a single command.

To verify that the Marketing section can't access any other services running on the Server, you can access the FTP service running on the Server from PC0. The request must be blocked by the ACL.



Updating the extended ACL

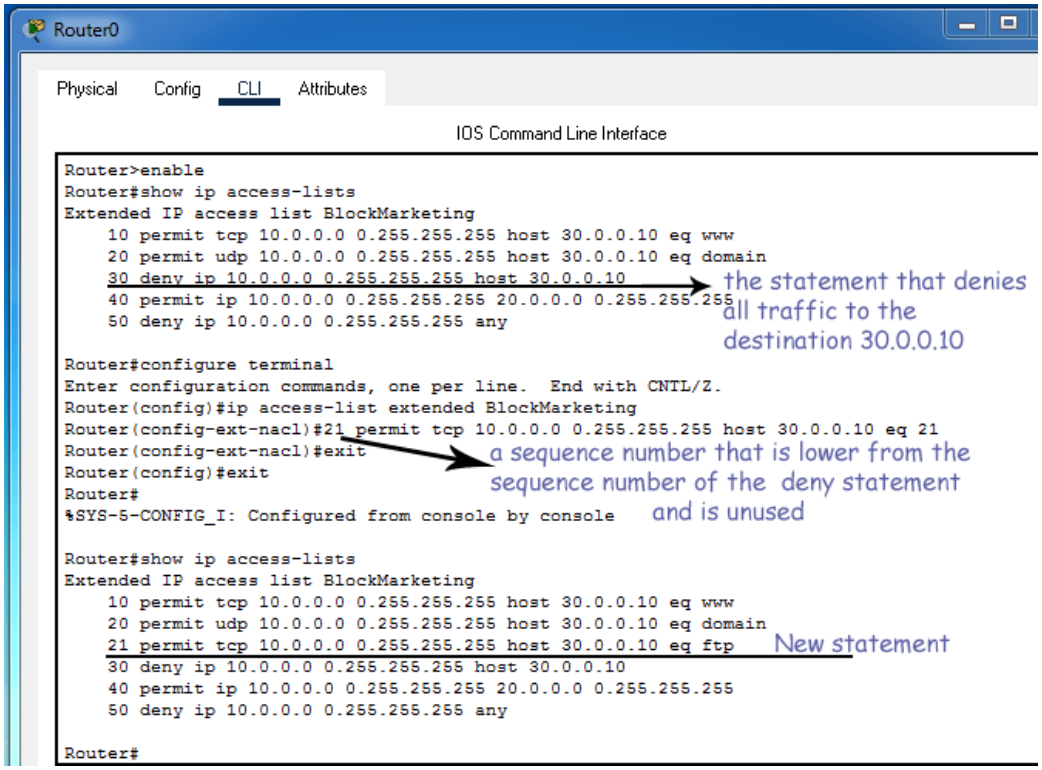
Now suppose, we want to allow the Marketing section to access the FTP service running on the Server. For this, we have to create an allow statement and will have to insert this statement before the statement that denies all traffic to the Server.

To view the sequence number of current statements, we can use the **'show ip access-lists'** command. Check the sequence number of the statement that denies all traffic to the destination 30.0.0.10. To insert a statement that allows FTP traffic, use a sequence number that is lower than the sequence number of the deny statement.

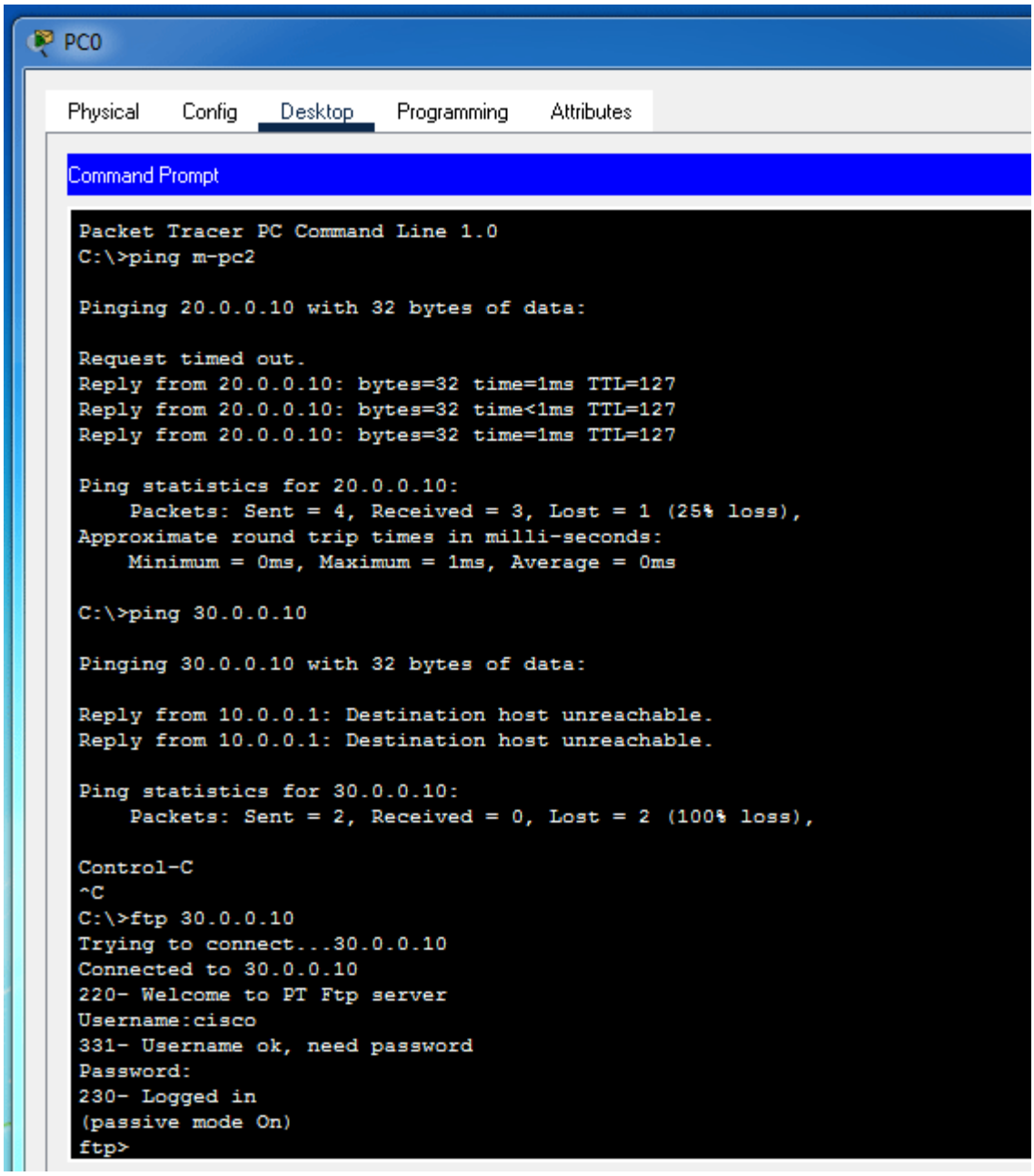
The following commands perform the above tasks.

```
Router>enable
Router#show ip access-lists
Extended IP access list BlockMarketing
10 permit tcp 10.0.0.0 0.255.255.255 host 30.0.0.10 eq www
20 permit udp 10.0.0.0 0.255.255.255 host 30.0.0.10 eq domain
30 deny ip 10.0.0.0 0.255.255.255 host 30.0.0.10
40 permit ip 10.0.0.0 0.255.255.255 20.0.0.0 0.255.255.255
50 deny ip 10.0.0.0 0.255.255.255 any
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ip access-list extended BlockMarketing
Router(config-ext-nacl)#21 permit tcp 10.0.0.0 0.255.255.255
host 30.0.0.10 eq 21
Router(config-ext-nacl)#exit
Router(config)#exit
Router#
Router#show ip access-lists
Extended IP access list BlockMarketing
10 permit tcp 10.0.0.0 0.255.255.255 host 30.0.0.10 eq www
20 permit udp 10.0.0.0 0.255.255.255 host 30.0.0.10 eq domain
21 permit tcp 10.0.0.0 0.255.255.255 host 30.0.0.10 eq ftp
30 deny ip 10.0.0.0 0.255.255.255 host 30.0.0.10
40 permit ip 10.0.0.0 0.255.255.255 20.0.0.0 0.255.255.255
50 deny ip 10.0.0.0 0.255.255.255 any
Router#
```

The following image shows how to run the above commands on the Router.



To verify that the Marketing section can access the FTP service running on Server. Open the command prompt on PC0 and access the FTP server running on Server. If PC0 can access the FTP server running on Server, it verifies that the ACL has been successfully updated for the new requirement.


PC0
Physical
Config
Desktop
Programming
Attributes
Command Prompt
Packet Tracer PC Command Line 1.0
C:\>ping m-pc2
Pinging 20.0.0.10 with 32 bytes of data:
Request timed out.
Reply from 20.0.0.10: bytes=32 time=1ms TTL=127
Reply from 20.0.0.10: bytes=32 time<1ms TTL=127
Reply from 20.0.0.10: bytes=32 time=1ms TTL=127
Ping statistics for 20.0.0.10:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>ping 30.0.0.10
Pinging 30.0.0.10 with 32 bytes of data:
Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Ping statistics for 30.0.0.10:
Packets: Sent = 2, Received = 0, Lost = 2 (100% loss),
Control-C
^C
C:\>ftp 30.0.0.10
Trying to connect...30.0.0.10
Connected to 30.0.0.10
220- Welcome to PT Ftp server
Username:cisco
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>