



Romel Cabling ▾



[Home](#)

[Home](#) > [My courses](#) > [Network Defense and Remote Access Configuration](#) > [O8 \[Enter Module Title Here\]](#) > [Lesson Proper for Week 8](#)

# Lesson Proper for Week 8

## Examining the Routing Process

Routing is the process of transporting packets of information across a network from the source node to the destination node. Routers determine the best path for the packet to take and then send the packet on its way. To determine the best path, routers use metrics such as hop count, bandwidth, or link state. However, the administrator can also configure predetermined paths for packets based on the protocols being transported and other variables. Routing takes place at the Network layer of the OSI (Open Systems Interconnection model).

While the routing process can be complicated, fundamentally the following steps take place:

1. An application, utility, or service on a source computer generates a packet to send to a specific destination. The OSI Network layer function of the source computer determines whether the destination is on the same network segment as the source computer or on a different one. If the destination is on the same network as the source, the packet is sent directly. If the destination is on a different network from the source, the packet is sent to the interface configured on the source as the default gateway—the interface on a router that gives a computer access outside its own network.
2. The router receives the packet and strips off the Data Link layer header, which includes source and destination MAC (media access control) addresses. Next, the router examines the destination Network layer address.
3. Based on the destination address and the information the router maintains in its routing table, the router determines which of its multiple interfaces to use to move the packet closer to its destination.
4. The router builds a header that is appropriate for the selected outbound interface. For example, if the outbound interface uses a token ring network, the router builds a token ring header so that the interface can understand the information and process the packet. The header contains Data Link layer source and destination addresses and protocol information.

5. The packet is sent through the selected interface to the next hop. This explanation of routing has been simplified. Typically, each packet requires the router to perform additional processing tasks, such as checking access control lists to filter traffic or tunneling different protocols. For example, an AppleTalk or IPX/SPX packet might be sent through an IP network. Generally, protocols do not “talk” to each other without an intermediary device to provide translation.

## **Router Security Fundamentals**

Because routers contain detailed information about network topology and addressing, they are a target for malicious attacks. Furthermore, one of the three main goals of information security—data availability—can be seriously compromised when routers do not perform their duties properly. Router security is an important component of network defense. Routers also play a role in defensive countermeasures when they work in conjunction with an IDPS to block packets from a source that the IDPS has determined to be a threat.

In this section, you learn about the function and configuration of access control lists used to perform packet filtering. You examine router logging and its importance in network security. Later in this chapter, you learn about router authentication, banners, and the use of Secure Shell to encrypt configuration connections.

## **Creating and Using Access Control Lists**

Router access control lists (ACLs) are permit or deny statements that filter traffic based on the source and destination address, source or destination port number, and protocol in the packet header. ACLs provide traffic-flow control and enhance network security. They can also be used to fine-tune performance and control client access to sensitive network segments.

### **Use and Rules**

You can approach ACLs in two ways: specify what traffic to deny and permit all other traffic, or specify what to permit and deny all other traffic. The latter approach is more secure, but it requires more planning and a more complex list. You must consider two factors when configuring ACLs:

- ACLs end with an implicit “deny any” statement, which means any packet that does not match the requirements for passage is blocked. Although this approach is more secure—and is the default behavior on Cisco ACLs—it might not serve the network’s needs and could block desirable traffic. To remedy this problem in networks when you do not need to block all access, include a “permit any” statement at the end of an ACL.
- ACLs are processed in sequential order. To avoid unnecessary use of router processing resources, rules that match common network traffic should be placed higher on the list so that they are processed first and the packet does not need to be compared to a large number of rules before the appropriate rule is identified for the packet. Careful planning is necessary to ensure that allowed packets are not blocked inadvertently, and packets that should be blocked do not slip through. Table 4-3 summarizes some of the problems you should be aware of when creating ACLs and their solutions.

Problem	Solution
Lack of planning results in simple logic mistakes.	Plan carefully what needs to be filtered and what needs access.
Sequential processing results in filtering errors.	Use the <i>IP Access List Entry Sequence Numbering</i> feature in Cisco IOS versions 12.2 and above, which allows you to move and insert rules in an existing ACL.
Applying ACLs via Telnet can result in lost connectivity for the administrator applying the list.	Use the reload command to restore access as long as the running configuration was not copied to the startup configuration.

Table 4-3 ACLs: Common problems and solutions

Remember the following general rules for ACLs:

- Routers apply lists sequentially.
- Packets are processed only until a match is made, and then they are allowed or denied.
- Lists always end with an implicit “deny any” statement.
- ACLs must be applied to an interface as inbound or outbound filters.
- The terms inbound and outbound refer to the perspective of the router; a packet entering the router is considered inbound, and a packet exiting the router is considered outbound.
- ACLs are not active until they are applied to an interface.
- Only one ACL per protocol and per direction can be applied to an interface.
- ACLs take effect immediately, but if you want the list to be permanent, you must copy the running configuration to the startup configuration using the copy running-config startup-config command.

Test ACLs thoroughly before applying them to a production router in your network; make sure they work as intended and correct any errors before you apply them. You should have a baseline for your network so that you know what “normal” traffic looks like. If you have a baseline before and after applying an ACL, you can determine where problems might occur. Ideally, you should have a baseline in your test network first.

## Standard ACLs

Standard ACLs have minimal configuration options. They can filter only on source IP address information, such as a host, subnet, or network address. Of course, there is no need to create a complex ACL if a simple one will do the job. Increasing complexity leads to increased chances for errors, so you should use extended ACLs only if you have a good reason for that level of control. Standard ACLs can be configured for IP, Internetwork Packet Exchange (IPX), AppleTalk, and other protocols. The most commonly used ACLs are for IP, so the next sections focus on it.

After configuring a standard ACL, you must apply it to the interface and specify the direction of the filter. ACLs are applied to inbound or outbound packets, and only one ACL per direction can be applied to an interface at a time.

## Standard IP ACLs

With standard IP ACLs, you can permit or deny traffic from a source host, a subnet, or an IP address. Destination addressing does not affect standard ACLs, which work well for simple packet filtering. Standard ACLs, like all varieties of ACLs, use an inverse mask that tells the router which bits in the address to be filtered are significant. (An inverse mask is indicated by the “source wildcard mask” parameter in commands.) A 0 bit in an inverse mask means to check the corresponding bit value in the IP address, and a 1 bit means to ignore the corresponding bit value in the IP address. An inverse mask of 0.0.0.0 means that all the bits are significant, so that specific IP address is filtered. To indicate the filtering of an entire subnet, mask only the host ID portion of the address. For example, the inverse mask to filter the entire 172.16.0.0 network, which has a default subnet mask, would be 0.0.255.255. The 0s are significant; the 255s are ignored.

*Remember that a 0 or a 1-bit value in an IP address refers to its binary format. For instance, the inverse mask 0.0.0.255 (decimal format) is the same as 00000000.00000000.00000000.11111111 (binary format).*

Standard ACLs have the following characteristics:

- They can filter based on source address.
- They can filter by host, subnet, or network address using an inverse mask.
- They should be placed on the router interface as close to the destination as possible.
- They have a default inverse mask of 0.0.0.0.

Standard ACLs use the following syntax:

```
access-list [list#] [permit|deny] [source IP address] [source wildcard mask]
```

The following list explains the parameters in this syntax:

- list#—Standard IP ACLs are represented by a number from 1 to 99.
- permit|deny—Specifies a permit or deny action to be taken when a packet is identified that meets the other filtering parameters.
- source IP address—Indicates the IP address of the source in the packet header to be identified for filtering.
- source wildcard mask—Determines which bits of the source address must match for the packet to be identified for filtering.

## Extended ACLs

Extended ACLs offer many more filtering options than a standard list. A standard list can filter based on source addressing, but an extended list can provide fine-tuned control over source and destination addresses, ports, and protocols that you want to filter. Of course, increased complexity means more chances of making a mistake, so be careful when creating and using extended ACLs. As with standard ACLs, extended ACLs are typically used in TCP/IP networks. However, you may need to set ACLs for IPX or other protocols.

## Extended IP ACLs

An extended IP ACL can filter based on source and destination address, port number, and protocol type. It uses the following syntax:

```
access-list [list#] [permit|deny] [protocol] [source IP address] [source wildcard mask] [operator] [port] [destination IP address] [destination wildcard mask] [operator] [port] [log]
```

The following list explains the parameters in this syntax:

- **list#**—Extended IP ACLs are represented by a number from 100 to 199.
- **permit|deny**—Specifies a permit or deny action to be taken when a packet is identified that meets the other filtering parameters; ACLs can have many lines specifying that a certain type of traffic is permitted or denied access.
- **protocol**—The IP protocol to be filtered; IP includes all protocols in the TCP/IP stack or a specific protocol, such as TCP (Transmission Control Protocol), UDP (User Datagram Protocol), IGMP (Internet Group Message Protocol), ESP (Encapsulating Security Payload), AHP (Authentication Header Protocol), GRE (Generic Routing Encapsulation Protocol), or ICMP (Internet Control Message Protocol).
- **source IP address**—Indicates the IP address of the source in the packet header to be identified for filtering.
- **source wildcard mask**—Determines which bits of the source address must match for the packet to be identified for filtering.
- **destination IP address**—Indicates the IP address of the destination in the packet header to be identified for filtering.
- **destination wildcard mask**—Determines which bits of the destination address must match for the packet to be identified for filtering.
- **operator**—Less than (lt), greater than (gt), or equal to (eq); operators are used if the ACL filters a port number or range of port numbers.
- **port**—The source or destination port number of the protocol, depending on the position of the port specification in the rule.
- **log**—Turns on logging of ACL activity.

You should remember the following points about extended IP ACLs:

- Extended IP ACLs do not have a default inverse mask of 0.0.0.0. An inverse mask must be specified for the source. (Keywords such as "host" can be used.)
- Extended IP ACLs should be applied to an interface as close to the traffic source as possible.
- The "established" parameter can be used to allow incoming traffic that responds to an internal request. For example, to allow the network 101.0.0.0 to receive DNS resolution or responses to other network service requests, enter the following line in the ACL:

access-list 100 permit tcp any 101.0.0.0 0.255.255.255 established

- Extended IP ACLs, like all ACLs, must be applied to an interface to be active. Also, remember that only one ACL per interface per direction can be active.
- There must be at least one permit access control entry in every access control list.

The following list explains the parameters in this syntax:

- ip access-list—This command is used to create the list name.
- type—This specifies the type of list; for example, you can specify extended or standard.
- name—This is the name to be assigned to the ACL.

The following example shows a command to create a named ACL with the name ResearchLAN. This process begins in global configuration mode: ip access-list extended ResearchLAN

## Examining Cisco Router Logging

Logging is a vital component of security because it provides information for troubleshooting, monitoring traffic patterns, and discovering and tracking down possible security incidents. Cisco routers can log a variety of events, and they use the following types of logging:

- AAA logging—Authentication, authorization, and accounting (AAA) logging collects information about remote user connections, commands issued, logons, logoffs, HTTP access, and similar events. AAA logs are sent to an authentication server by using the Terminal Access Controller Access Control System Plus (TACACS+) protocol, the Remote Authentication Dial-In User Service (RADIUS) protocol, or a combination of both.
- SNMP trap logging—Simple Network Management Protocol (SNMP) trap logging sends notifications of system status changes to SNMP management stations. This logging method is normally used with an existing SNMP infrastructure.
- System logging—Depending on the system configuration, system logging reports system logs to different locations, including the system console port, UNIX servers via the syslog protocol, or a local logging buffer in router RAM.

Log events can also be monitored via remote sessions by using virtual terminal (VTY), or TeleTypewriter or text telephone (TTY) lines. You can gain remote access to the router command line in several ways, but all inbound connections are made with TTY lines. The most important security events recorded by system logging are changes to the system configuration, ACL matches, interface status changes, and optional firewall or IDPS events.

**Logging Levels** Logging events are tagged with an urgency or severity level ranging from 0 to 7, with 0 indicating the highest urgency and 7 the lowest. Table 4-4 lists these severity levels.

Logging destinations can be configured with a severity level so that events below that level are not recorded. If your router is logging all levels, your log files fill up quickly, increasing the risk of overwriting critical information. You can specify various destinations for logging, and you can buffer and view logging messages by using the show logging command at the privileged exec mode prompt.

This method has disadvantages, however. First, you must be at the terminal or access it remotely to view logs. Second, and more importantly, buffered logging is limited by the amount of memory in the router. Large log files cannot be stored in the router's memory buffer without the risk of serious performance problems and lost logs.

Level	Urgency
0	Emergency—system is unusable
1	Alert—requires immediate action
2	Critical—indicates a critical condition
3	Error—indicates an error condition
4	Warning—specifies a warning condition
5	Notification—indicates a normal but possibly significant condition
6	Informational—displays an informational message
7	Debugging—displays a debugging message

Table 4-4 Cisco router logging severity levels

Logging Options for larger log files, you can use a syslog server. Most Windows, UNIX, or Linux servers can be configured to host router logs. Use the logging host command to direct a router to send logs to a specific location.

Figure 4-3 shows options for the logging command. As you can see, a number of options are available for logging destinations and severity levels. You can specify the name or IP address of the logging host and set logging levels for the console, terminal lines, or syslog server.

```

Branch03(config)#logging ?
  Hostname or A.B.C.D  IP address of the logging host
  buffered             Set buffered logging parameters
  buginf              Enable buginf logging for debugging
  cns-events          Set CNS Event logging level
  console             Set console logging parameters
  count              Count every log message and timestamp last occurrence
  esm                Set ESM filter restrictions
  exception           Limit size of exception flush output
  facility            Facility parameter for syslog messages
  filter              Specify logging filter
  history             Configure syslog history table
  host               Set syslog server IP address and parameters
  monitor            Set terminal line (monitor) logging parameters
  on                 Enable logging to all enabled destinations
  origin-id          Add origin ID to syslog messages
  queue-limit        Set logger message queue size
  rate-limit         Set messages per second limit
  reload             Set reload logging level
  server-arp         Enable sending ARP requests for syslog servers when
                    first configured
  source-interface   Specify interface for source address in logging
                    transactions
  trap              Set syslog server logging level
  userinfo           Enable logging of user info on privileged mode enabling

```

Figure 4-3 Options for the logging command

*Syslog is a simple protocol that sends a small text message via UDP or TCP to the server hosting the logs. Syslog was originally developed for UNIX and Linux systems, but variations for other systems are available. A detailed discussion is beyond the scope of this book, but you can learn more by reading RFCs 3164 and 3195.*

Buffered Logging You can set parameters for buffered logging, which stores log output files in the router's memory (RAM). Although buffered logging has limitations, it is useful for troubleshooting purposes. Figure 4-4 shows options for the logging buffered command.

```

Branch03(config)#logging buffered ?
  <0-7>              Logging severity level
  <4096-2147483647> Logging buffer size
  alerts            Immediate action needed           (severity=1)
  critical          Critical conditions                (severity=2)
  debugging         Debugging messages               (severity=7)
  emergencies       System is unusable                (severity=0)
  errors           Error conditions                   (severity=3)
  filtered          Enable filtered logging
  informational     Informational messages            (severity=6)
  notifications     Normal but significant conditions (severity=5)
  warnings          Warning conditions                 (severity=4)
  xml              Enable logging in XML to XML logging buffer
  <cr>

```

Figure 4-4 Options for the logging buffered command



Antispoofing Logging Antispoofing is a way to prevent spoofing and ensure that no packets arrive at your security perimeter with a source address of your internal network or certain well-known or reserved addresses. Antispoofing is accomplished by using ACLs. Because ACLs are so vital to security and are the primary means of implementing basic security on a router, you learn more about them in this section.

Your ACL should instruct the router to deny any inbound packet with a source address that matches your internal network, broadcast, and loopback addresses; illegal addresses, such as all 0s or all 1s; and multicast or experimental address classes. At the end of each rule in the ACL, specify that packets matching these conditions will be logged, as shown in this example: deny any 172.16.0.0 0.0.255.255 any log

By adding the log keyword to the end of an extended ACL, you tell the router to send information about matching packets to the router's log. When configuring the ACL, you can specify a remote logging host, as shown in the following example, in which a named ACL is created to provide antispoofing based on the parameters discussed earlier in this section:

```
ip access-list extended ResearchLAN

remark Antispoofing ACL

deny icmp any any redirect

deny ip 180.50.0.0 0.0.255.255 any log

deny igmp 224.0.0.0 31.255.255.255 any

permit ip any any log

exit

logging 180.50.0.12

interface FastEthernet 0/0

ip access-group ResearchLAN in
```

In this example, a named ACL, ResearchLAN, is created. A notation is made in the ACL to help identify its purpose. Attackers often use ICMP redirects to disrupt the function of a router, but the first access control entry (ACE) denies these packets. The second ACE denies and logs any packets that attempt to enter the network if they “pretend” that they come from inside the network. (This network has an address of 180.50.0.0/16.) The third ACE blocks IPv4 multicasts, and the last ACE allows all other packets; however, these packets are logged as well. The logging command specifies the IP address of the computer that hosts the log files, and the last two lines in the preceding example link the ACL to the Ethernet 0 interface, specifying that inbound packets will be filtered.

*You may be wondering about the difference between the access-list and ip access-group commands. The access-list command creates the ACL. The ip access-group command assigns an ACL to an interface and specifies the direction, inbound or outbound.*

Once the ACL is created and applied to an interface, you can use the `show ip access-lists` command from privileged exec mode to review the ACLs, as shown in Figure 4-5. Note that lines are preceded with numbers that are incremented by 10. These numbers allow you to modify or insert ACEs without having to reconstruct the entire ACL from scratch.

```
Branch03#show ip access-lists
Extended IP access list ResearchLAN
 10 deny icmp any any redirect
 20 deny ip 180.50.0.0 0.0.255.255 any log
 30 deny igmp 224.0.0.0 31.255.255.255 any
 40 permit ip any any log
```

Figure 4-5 Output of the `show ip access-lists` command

## Cisco Authentication and Authorization

Authentication and authorization on a router work much like they do with a server. They identify users and allow or deny access based on the users' credentials. Authentication is the process of determining that users are who they say they are. Authorization specifies what users are allowed to do after they have accessed the system. For example, Bob is allowed to authenticate to the router using his username and password to check an interface's status, but he is not authorized to change settings on the interface.

The two types of authentication on a Cisco router are AAA and non-AAA. AAA, as mentioned previously, stands for authentication, authorization, and accounting. Cisco's AAA architecture uses one or more of three security protocols to enhance security: TACACS+, RADIUS, and Kerberos. TACACS+ is a proprietary Cisco protocol that uses TCP (port 49) for transport and encrypts all data. It also allows multiple levels of authorization and can use other authentication methods. RADIUS is an open standard that uses UDP (ports 1812 and 1813) and encrypts only passwords.

Although AAA is the recommended method for access control, you can also use non-AAA methods, such as local username authentication or enable password authentication. Any method that does not use Cisco AAA Security Services is considered non-AAA. The following sections explain passwords that you can configure on Cisco routers.

### Router Passwords

Passwords are the main defense against attacks on your router, but remember that if attackers have physical access to your router, they can access all its configuration settings. Cisco routers have five types of passwords you must be able to configure:

- Enable
- Enable secret
- AUX

- VTY
- Console

Because a router's main purpose is to connect networks, you might see Ethernet, token ring, Fibre Distributed Data Interface (FDDI), T1, T3, ATM (Asynchronous Transfer Mode), or other interface modules. Many Cisco routers are modular, so you can customize them by selecting the interface modules you want and specifying the encryption levels and RAM that your router must have. As mentioned, setting AAA security is the preferred method of securing access to the router; however, you still need to have local access to privileged mode in case of router or network problems.

Before setting passwords, you must know some requirements. Passwords must be 1 to 25 characters long. Leading spaces in the password are ignored, but any other spaces in it are considered part of the password. Also, the first character cannot be a number. In addition, Cisco passwords have three levels of encryption: type 0, which provides no encryption; type 7, which is encrypted but can be decrypted by router password-cracking tools that are readily available on the Internet; and the strongest level, type 5, which is a Message Digest 5 (MD5) hash. Although MD5 is a one-way hash and cannot be decrypted, it is still susceptible to brute-force attacks. Because of the relative weakness of this password system, it is important to audit the use and configuration of routers.

#### Enable Passwords

The enable password's main purpose is to prevent casual or accidental access to privileged exec mode. Because it uses weak encryption, it provides no real security against more determined and knowledgeable attackers. Cisco recommends using an enable secret password instead of an enable password.

#### Enable Secret Passwords

The enable secret password uses type 5 encryption and overrides an enable password. The following line shows you how to set an enable secret password:

```
Branch06(config)#enable secret Pa33m04d Remember that no password can defend against intrusion if an attacker can physically access your router; resetting a password on a router is simply too easy.
```

**AUX Passwords** Normally, the auxiliary port on a router is connected to a modem to allow remote access for router configuration.

Configuring an AUX password is much like configuring a console password:

```
Branch06(config)#line aux 0
```

```
Branch06(config-line)#password M0d3m
```

```
Branch06(config-line)#login
```

For more information, see “Console Passwords” later in this section.

**VTY Passwords** Most Cisco routers support up to five simultaneous VTY sessions; by default, no passwords are assigned to these sessions. Cisco’s built-in security on VTY lines requires configuring passwords to access the router through a VTY session, usually with Telnet. The following example shows how to set all five lines at once, although you can set each line separately if you want:

```
BranchO6(config)#line vty 0 4
```

```
BranchO6(config-line)#password Kodiak
```

```
BranchO6(config-line)#login
```

### Console Passwords

You use the console port to directly connect a router to a laptop or other computer using a program such as HyperTerminal or PuTTY. The console port is normally used to set up a new router, but it can also be used to reset a router’s password. For this reason, physical control of the router is critical. For security reasons, some organizations prohibit any router configuration except from the console port, eliminating the chance that transmissions can be sniffed. The following example shows how you might configure a console password:

```
BranchO6(config)#line console 0
```

```
BranchO6(config-line)#password J3N1ff54
```

```
BranchO6(config-line)#login
```

```
BranchO6(config-line)#end
```

### Encrypting Passwords

By default, the enable secret password is the only encrypted password type. By using the service password-encryption command in global configuration mode, as shown in the following example, you encrypt all passwords on the router using reversible encryption:

```
BranchO6(config)#service password-encryption
```

You can verify the password encryption state by using the show running-configuration command from privileged exec mode. Figure 4-6 shows part of the command output on a system in which all passwords were set to “Pa\$\$word” but the service password-encryption command was not entered.

*It is unwise to set all passwords to be the same. Furthermore, “Pa\$\$word” is a weak password because it is based on a word in the dictionary, and password-cracking programs know that “\$” can mean “s.”*

```
line con 0
password Pa$$word
login
line aux 0
password Pa$$word
line vty 0 4
password Pa$$word
```

Figure 4-6 Unencrypted passwords in the show running-configuration command output

Figure 4-7 shows the same output after the service password-encryption command has been executed. The number “7” before the encrypted passwords indicates the level of encryption used.

```
line con 0
password 7 03345A4F421B2E5E4A
login
line aux 0
password 7 08114D0A4D0E0A0516
line vty 0 4
password 7 08114D0A4D0E0A0516
```

Figure 4-7 Encrypted passwords in the show running-configuration command output

## Banners

Banners are messages displayed to greet users who log on to a router. Banners provide information or warnings during logons, during privileged exec mode processes, or for an incoming asynchronous line connection. The most common banners display legal disclaimers and warnings at logon, and all banners should include a legal warning that clearly states the company's policy on unauthorized access.

*Legal disclaimers can cover complex issues, so legal counsel should be consulted to write or approve the disclaimers if your budget allows it. You need to address different jurisdictions because they might require varying notifications, particularly when transmissions are monitored.*

A typical banner might look like this:

WARNING! Authorized access only. Unauthorized access prohibited. This system is the property of [Company Name]. Disconnect IMMEDIATELY if you are not an authorized user!

A banner might be more explicit. For example, the following banner is modified from the U.S. Department of Energy's Computer Incident Advisory Capability (CIAC):

THIS IS A PRIVATE COMPUTER SYSTEM. It is for authorized use only. Users (authorized or unauthorized) have no explicit or implicit expectation of privacy.

Any or all uses of this system and all files on this system may be intercepted, monitored, recorded, copied, audited, inspected, and disclosed to authorized site and law enforcement personnel as well as authorized officials of other agencies, both domestic and foreign. By using this system, the user consents to such interception, monitoring, recording, copying, audition, inspection, and disclosure at the discretion of authorized site personnel.

Unauthorized or improper use of this system may result in administrative disciplinary action and civil and criminal penalties. By continuing to use this system you indicate your awareness of and consent to these terms and conditions of use. LOG OFF IMMEDIATELY if you do not agree to the conditions stated in this warning.

A banner should never include wording that could give attackers information about your system or network, such as names, IP addresses, and software versions. Avoid displaying any information that an attacker might seek in network reconnaissance.

◀ Preliminary Activity for Week 8

Jump to...



Analysis, Application, and Exploration for Week 8 ▶



## Navigation

Home



Dashboard

Site pages

My courses

Capstone Project 2

Network Defense and Remote Access Configuration

Participants

General

06 - Preliminary Examination

07 [Enter Module Title Here]

08 [Enter Module Title Here]



Preliminary Activity for Week 8



**Lesson Proper for Week 8**



Analysis, Application, and Exploration for Week 8



Generalization for Week 8



Evaluation for Week 8



Assignment for Week 8

## Fair Warning





**NOTICE:** Please be reminded that it has come to the attention of the Publishing Team of eLearning Commons that learning materials published and intended for *free use only by students and faculty members within the eLearning Commons network were UNLAWFULLY uploaded in other sites without due and proper permission.*

**PROSECUTION:** Under Philippine law (Republic Act No. 8293), copyright infringement is punishable by the following: Imprisonment of between 1 to 3 years and a fine of between 50,000 to 150,000 pesos for the first offense. Imprisonment of 3 years and 1 day to six years plus a fine of between 150,000 to 500,000 pesos for the second offense.

**COURSE OF ACTION:** Whoever has maliciously uploaded these concerned materials are hereby given an ultimatum to take it down within 24-hours. Beyond the 24-hour grace period, our Legal Department shall initiate the proceedings in coordination with the National Bureau of Investigation for IP Address tracking, account owner identification, and filing of cases for prosecution.

---

## Activities

-  Assignments
-  Forums
-  Quizzes
-  Resources