



Romel Cabling ▾



[Home](#)

[Home](#) > [My courses](#) > [Network Defense and Remote Access Configuration](#) > [07 \[Enter Module Title Here\]](#) > [Lesson Proper for Week 7](#)

# Lesson Proper for Week 7

## Examining the Common Vulnerabilities and Exposures Standard

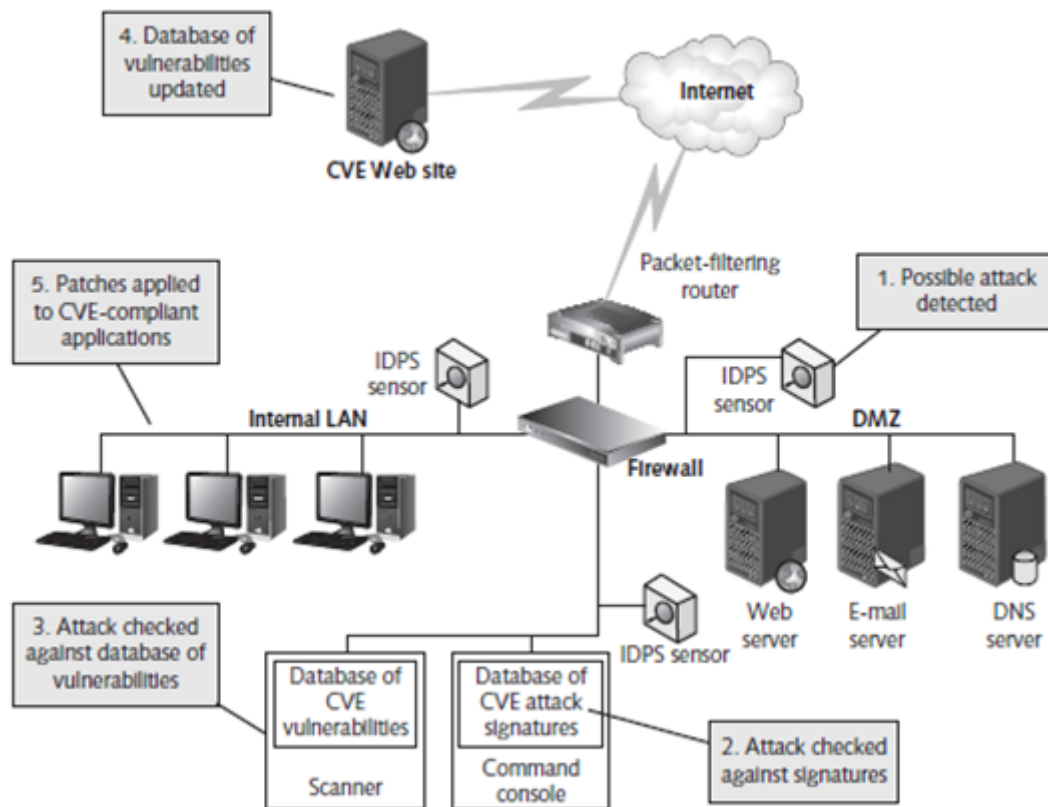
One way to prevent attacks is to make sure your security devices can share information and coordinate with one another. At the perimeter of any network, you are likely to have a variety of hardware and software devices that provide security and that need to work cooperatively with one another. You might have a router from one vendor, a firewall from another, and an IDPS from a third. Unfortunately, the way these devices interpret signatures might differ. They probably address the same known attacks but name them differently and describe their characteristics differently. The Common Vulnerabilities and Exposures (CVE) standard enables these devices to share information about attack signatures and other vulnerabilities so that they can work together.

## How CVE Works

CVE enables hardware and security devices that support it to draw from the same databases of vulnerabilities, which are presented in a standard format. For instance, a scanner is a device that scans a network for open ports or other potential vulnerabilities. If the scanner supports CVE, you can use it to compile a report that lists weak points in the system. When an alarm message is transmitted by an IDPS that also supports CVE, the attack signature can be compared to the report of current vulnerabilities to see whether an attack has actually occurred.

As you can see in Figure 3-1, the CVE standard affects many parts of a network:

1. A possible attack is detected by an IDPS sensor.
2. The signature is checked for a match against the database of known attack signatures available to the IDPS. If the IDPS being used is also CVE compliant, the report on the attack contains information on known network vulnerabilities associated with the attack signature.



**Figure 3-1 CVE enables multiple devices to work together to detect possible attacks**

3. The list of vulnerabilities is compared against a database of current vulnerable points in the system that have been compiled and stored by a CVE-compliant scanner to determine whether this possible attack can have an impact on the network.
4. Periodically, the list of vulnerabilities is updated with new entries from the CVE vulnerability Web site.
5. The manufacturers of CVE-compliant applications generate patches and updates in response to vulnerabilities. Those patches can then be applied to applications on the network.

Great benefits, such as stronger security and better performance, result when all the security devices on a network understand and use information that complies with the CVE standard. If you are responsible for purchasing an IDPS and other equipment for your organization, you should make sure they support CVE.

### Understanding Signature Analysis

A signature is a set of characteristics—such as IP numbers and options, TCP flags, and port numbers—that define a type of network activity. Besides individual TCP/IP packet attributes, a signature can also consist of a sequence of packets or other events, such as logons to a network.

Some intrusion-detection devices assemble databases of “normal” traffic signatures. As traffic is detected, it is compared to the database, and any deviations from normal signatures (the network baseline) trigger an alarm. Intrusion-prevention devices can also go further and drop the packets, and perhaps create firewall filtering rules to prevent similar traffic from entering the network. Other intrusion-detection devices refer to a database of well-known attack signatures. Any traffic that matches one of the stored attack signatures triggers an alarm or causes the IDPS to take steps to prevent the attack. Your understanding of normal and suspicious traffic signatures enables you to configure an IDPS to work more effectively by minimizing the number of false positives (false alarms) and decreasing the number of false negatives (missed attacks). You learn about false positives and false negatives in Chapter 8.

The following sections introduce you to signature analysis. Then, you learn more about analyzing packets and review normal traffic signatures you are likely to encounter. Finally, you learn about suspicious traffic signatures that indicate a possible attempt to scan and gain unauthorized access to your network.

Signature analysis is the practice of analyzing and understanding TCP/IP communications to determine whether they are legitimate or suspicious. Packets are the most basic level of network communications.

Suspicious TCP/IP packets fall into several categories: bad header information, suspicious data payload, single-packet attacks, and multiple-packet attacks.

### **Bad Header Information**

Packets are commonly altered through their header information, and packet filters usually scan for these alterations. Suspicious signatures can include malformed data that affect some or all of the following:

- Source and destination IP address
- Source and destination port number
- IP options
- IP fragmentation flags, fragmentation offset, or fragment identification
- IP protocol
- IP, TCP, or UDP checksums

A checksum is a simple error-checking procedure for determining whether a message has been tampered with or damaged in transit. A mathematical formula is used to process the number of data bits in a message. A numeric value (the checksum) is then calculated. The receiving computer applies the same formula to the message; if a different checksum is found, the receiving computer determines that the message has been tampered with or corrupted and drops it.

Attackers can use software that generates packets set to their specifications to forge IP addresses or other types of header information. For instance, a packet can be broken into “chunks” and sent in a series. An attacker can eliminate the initial chunk in the series from the set, which makes the receiving computer unable to reassemble the packets. Therefore, the series of chunks that follow can circumvent a packet filter. An attacker can send more or fewer packets than indicated in the initial packet, which could disable a server that cannot process a different number of packets than it expected to receive.

### **Suspicious Data Payload**

The payload (data) part of a packet is the actual data sent from an application on one computer to an application on another. Sometimes, attacks can be detected by an IDPS that matches a text string to a specific set of characters in the payload. For instance, a program called CyberEYE creates remote-access Trojans (RATs); when installed on unsuspecting systems, these RATs open back doors that give the remote attacker administrative rights on the victim’s computer. The CyberEYE program was created in Turkey, and one of its signatures is the presence of the following hexadecimal string in the data portion of the packet: 41 4E 41 42 49 4C 47 49 7C. When converted into ASCII, this string becomes ANA BILGI, which is Turkish for MAIN INFORMATION, according to security analyst Chris Sanders. The presence of this hexadecimal string in the data portion of the packet is a key part of the CyberEYE session establishment detection signature, particularly when the packet is communicated over TCP port 4433, a port frequently used by CyberEYE.

In another type of attack, the UNIX Sendmail program is exploited by adding codes to packet contents. Codes such as VRFY and EXPN are used to uncover account names on the Sendmail server. By adding the code EXPN DECODE in a packet’s data payload, attackers attempt to establish a connection with an alias called “decode.” If a connection is made, attackers can use it to place malicious files on the exploited system. To defend against this type of attack, a network administrator should remove the “decode” alias line, which is installed by default with many UNIX/Linux systems in the /etc/mail/aliases file.

*Internet searches can yield a wealth of tools and information, but be careful when downloading files. You might end up downloading the malicious software you are trying to avoid! Create a folder to store the downloaded files, unzip the contents into the new folder, if necessary, and then run a virus scan on the folder. Any known malware signature will be recognized, provided that you keep your antivirus software updated, and you can safely remove the files without harm in most cases. You can reduce risk by using common sense and basic security protocols.*

### **Single-Packet Attacks**

A single-packet attack (also called an “atomic attack”) can be completed by sending a single network packet from a client to a host. Because only one packet is needed, a connection does not need to be established between the two computers. Many changes to IP option settings can cause a server to freeze up because it does not know how to handle these packets. The IP option settings are shown in Table 3-1.

Option number	Name of option
0	End of Options
1	No Operation
2	Security
3	Loose Source and Record Routing
4	Internet Timestamp
7	Record Return Route
8	Option has been deprecated
9	Strict Source and Record Routing

**Table 3-1 IP option settings**

As an example of IP options processing, suppose an ICMP echo request (or “ping”) packet is sent from a host to a server with Option 7 set. The echo reply response from the server might spell out the route the request takes to return from the server, thus revealing the IP addresses of hosts or routers on the network that the attacker can then target. Option 4 can be used with Option 7 to record the amount of time the echo reply packet spends between “hops” on the network. A hop is the movement of a packet from one point on a network to another.

### **Multiple-Packet Attacks**

In contrast to single-packet attacks, multiple-packet attacks (also called “composite attacks”) require a series of packets to be received and executed. These attacks are especially difficult to detect. They require an IDPS to have multiple attack signatures on hand for reference. In addition, the IDPS sensor needs to maintain state information about a connection after it has been established, and it needs to keep that state information on hand for the entire length of an attack.

Denial of service (DoS) attacks are obvious examples of multiple-packet attacks. A type of DoS attack called an ICMP flood occurs when multiple ICMP packets are sent to a single host on a network. The result of this flood is that the server becomes so busy responding to the ICMP requests that it cannot process other traffic.

### **Analyzing Packets**

A packet sniffer captures information about each TCP/IP packet it detects. By using such an application, you can study packets and identify characteristic features that tell you what type of connection is under way and whether the transmission is legitimate or suspicious. Capturing packets and studying them can help you better understand what makes up a signature

### **Analyzing Traffic Signatures**

Now that you have learned the basics of packet captures, you must learn to determine whether traffic is normal or suspicious. You are probably familiar with the concept of network baselining. You must first know what is normal for your network before you can identify anomalies. This section helps you learn to tell the difference between normal traffic and suspicious activity.

## **FTP Signatures**

If your organization operates a public FTP server, you will be called on regularly to review the signatures of packets that attempt to access that server. You need to determine whether the computer that makes the connection attempt is allowed to access the server in accordance with your packet-filtering rules.

## **Web Signatures**

Most of the signatures you see in the log files you analyze will probably be Web-related. When a signature is Web-related, it consists of packets sent back and forth from a Web browser to a Web server as a connection is made. The signature of a normal handshake between two Web browsers consists of a sequence of packets that are distinguished by their TCP flags.

## **Examining Abnormal Network Traffic Signatures**

As IDPSs become more sophisticated, the techniques that attackers use to circumvent them have multiplied and become more complicated. Features such as illegal combinations of TCP flags and private IP addresses in packets are relatively easy to identify as abnormal, compared with attacks that use a range of packets. Suspicious traffic signatures can fall into one of the following categories:

- **Informational**—This traffic might not be malicious, but it could be used to verify whether an attack has been successful. Examples include ICMP echo request packets or TCP packets sent to a specific port on a specific system.
- **Reconnaissance**—This traffic could represent an attacker's attempt to gain information about a network as a prelude to an attack. Examples include ping sweeps and port scans.
- **Unauthorized access**—This traffic might be caused by someone who has gained unauthorized access to a system and is attempting to retrieve data from it. Examples include the Shixploit attack and the Hydraq attack.
- **Denial of service**—This traffic might be part of an attempt to slow or halt all connections on a network device, such as a Web server or mail server. ARP cache poisoning is an example of an attack that can cause denial of service.

Some of the more common examples of suspicious traffic—ping sweeps, port scans, random back door scans, and Trojan scans—are described in the following sections along with their signatures.

## **Ping Sweeps**

To gain access to specific resources on an internal network, a hacker needs to determine the location of a host. One method is to conduct a ping sweep (also called an ICMP sweep), which sends a series of ICMP echo request packets in a range of IP addresses. Usually, the messages come in quick succession. Multiple packets can be detected in a single second, indicating that an automated tool is being used.

Port Scans

If an attacker can determine any legitimate IP addresses on an internal network, the next step is to target one of those addresses and perform a port scan—an attempt to connect to a computer’s ports and check whether any are active and listening. An attacker who finds an open port can exploit any known vulnerabilities associated with any service that runs on that port. Many newer applications perform dynamic port negotiation based on available resources. However, vulnerabilities still exist in older applications and services.

Random Back Door Scans

You can think of a port as a virtual door through which data can enter and leave a computer. In that context, a back door is an undocumented or unauthorized hidden opening (such as a port) through which an attacker can access a computer, program, or other resource. One type of port scan probes the same computer ports used by well-known Trojan programs to see if any ports are open and listening. These applications seem to be harmless, but they can damage a computer or the files on it.

Specific Trojan Scans

Attackers can execute port scans in several ways. In a vanilla scan, all ports from 0 to 65,535 are probed one after another. In a strobe scan, an attacker scans only ports that are commonly used by specific programs in an attempt to see whether the program is present and can be used. Table 3-2 shows some examples of Trojans and the protocols and ports associated with them.

Trojan	Protocol	Port
Trojan.Asprox	TCP	80, 82
W32.Spybot.pen	TCP	8076
w32.myto <b>b</b> .jw@mm	TCP	10027
Trojan.Mitglieder.h	TCP	14247
Sub-7 2.1	TCP	27573
Remote Windows Shutdown	TCP	53001
Back Orifice 2000	UDP	54321

Table 3-2 Examples of Trojan programs and ports

**Nmap Scans Network Mapper (Nmap)** is a popular software tool for scanning networks. You should be able to recognize the common types of scans it enables attackers to perform. With Nmap, attackers can send packets that circumvent the normal three-way handshakes two computers use to establish a connection. Nmap enables attackers to send packets for which an IDPS might not be configured to send an alarm. The IDPS might see a combination of TCP flags that it does not recognize, and because no rule exists for the combination, an alarm might not be triggered.

## **Packet Header Discrepancies**

Discrepancies you find in TCP, IP, ICMP, or UDP packet headers can provide warning signs that an attacker has crafted the packet (in other words, manufactured or altered it on purpose). However, instead of seeing these discrepancies in a well-defined and lengthy succession of packets, you might receive only a single packet with a falsified IP address, falsified port number, illegal TCP flags, illegal TCP or IP options, or fragmentation abuses.

### **Falsified IP Address**

Your IDPS might send alarms for violations of IP header settings, as specified in RFC 791, “Internet Protocol.” For example, an IP address should not appear in one of the three reserved ranges (10.0.0.0 to 10.255.255.255, 172.16.0.0 to 172.31.255.255, and 192.168.0.0 to 192.168.255.255). The use of addresses in the reserved ranges is limited to private networks. If you see one of these addresses in a packet coming from the public network, a router or other device might have been misconfigured or is malfunctioning. On the other hand, the packet might show a private address because an attacker has used IP spoofing. In other words, the attacker has inserted a false address into the IP header to make the packet more difficult to trace back to its source.

**Falsified Port Number or Protocol** You already know that IP address information can be falsified in a packet, but protocol numbers might also be altered to elude an IDPS. TCP and UDP headers should never have the source or destination port set to 0 because this number is reserved by IANA. Protocol numbers are assigned by IANA, like port numbers. The Protocol field in IPv4 is an 8-bit field used to specify this number. (In IPv6, the field is called Next Header.) This field indicates the transport protocol to be used, such as ICMP (#1), TCP (#6), or UDP (#17). Currently, the number in this field cannot be set greater than 142. The numbers 143–252 are unassigned, 253 and 254 are used for experimentation and testing, and 255 is reserved. The use of undefined protocol numbers might indicate an attacker’s attempt to establish a proprietary communications channel—a channel known to one person and used only by that person.

**Illegal TCP Flags** As you saw in the section called “Examining Normal Network Traffic Signatures” earlier in this chapter, the TCP flags SYN and ACK are exchanged to establish a connection between two computers. The PSH flag is used when data is being sent, and the FIN flag is used when a connection is complete. Other normal TCP flag rules include the following:

- Every packet in a connection should have the ACK bit set, except for the initial SYN packet and possibly an RST packet used to terminate a connection.



- Packets during the “conversation” portion of the connection contain just an ACK flag by default. This portion of the connection occurs after the three-way handshake but before teardown or termination. Optionally, these packets can contain PSH and/or URG flags.
- FIN/ACK and ACK are used during the normal teardown of an existing connection. PSH, FIN, and ACK might also be seen near the end of a connection.
- RST or RST/ACK can be used to terminate a connection immediately.

One of the most obvious ways to detect an abnormal packet signature is to look at the TCP flags for violations of normal usage. A packet with the SYN and FIN flags set should not exist in normal traffic; however, an attacker might set both flags to cause the destination computer to crash or freeze because it does not know how to respond. After the server is disabled, the attacker can attack a computer on the internal network using an IP address detected earlier through network scans.

The following list summarizes signatures of malformed packets that misuse the SYN and

FIN flags:

- SYN FIN is probably the best-known illegal combination. Because SYN is used to start a connection and FIN is used to end one, it does not make sense to include both flags together in a packet. Many scanning tools use SYN FIN packets because older intrusion-detection systems often were not configured to recognize or block them. However, most IDPS devices are now configured to catch such illegal combinations. You can safely assume that hackers created any SYN FIN packets you see.
- Other variants of SYN FIN exist, including SYN FIN PSH, SYN FIN RST, and SYN FIN RST PSH. Their use is sometimes called an Xmas attack. These packets can be used by attackers who know that IDPSs might be looking for packets with just the SYN and FIN flags set. Packets should never contain a FIN flag by itself. FIN packets are frequently used for port scans, network mapping, and other stealth activities.
- A SYN-only packet, which should occur only when a new connection is being initiated, should not contain any data.

You might also encounter null packets—TCP packets with no flags set, which could cause a server to crash. It is a violation of TCP rules to use a packet with no flags set.

## TCP or IP Options

TCP options in a packet can alert you to intrusion attempts and even enable you to identify the type of OS being used. For instance, only one MSS or window option should appear in a packet. MSS, NOP, and SackOK should appear only in packets that have the SYN and/or ACK flag set.

IPv4 options were originally intended as ways to insert special handling instructions into packets. However, attackers mostly use IP options now for attack attempts. Because of this vulnerability, many filters simply drop all packets with IPv4 options set. In IPv6, the options field is removed and replaced by extension headers.

## Fragmentation Abuses

Every type of computer network, such as Ethernet, FDDI, and Token Ring, has its own maximum transmission unit (MTU)—the maximum packet size it can transmit. Packets that are larger than the MTU must be fragmented, or broken into multiple segments that are small enough for the network to handle.

After a packet is broken into fragments, each fragment receives its own IP header. However, in IPv4, only the initial packet in a set includes a header for higher-level protocols. Most filters need the information in the higher-level protocol header to make the decision to allow or deny the packet. Accordingly, attackers send only secondary fragments, which are any fragments other than the initial one. These packets are often allowed past the IDPS because filter rules are applied to first fragments only. IPv6 addresses this vulnerability in two ways: It permits only the source node to fragment payloads, and it divides the packet into unfragmentable and fragmentable parts. The fragmentable part can be processed only when the packet reaches its destination. The unfragmentable portion includes the IPv6 header, the hop-by-hop options header, the destination options header, and the routing header. This portion is processed by each router along the route to the destination.

Fragmentation can occur normally. However, an IDPS should be configured to send an alarm if it encounters a large number of fragmented packets. Many different types of fragmentation abuses can occur. Some of the more serious abuses are described briefly in the following list:

### IPv4

- Overlapping fragments—Two fragments of the same packet have the same position within the packet, so the contents overlap. A properly configured firewall should always drop this type of packet.
- Fragments that are too large—An IP packet can be no larger than 65,535 bytes. If packets are more than the maximum size when they are reassembled from their fragments, they might cause some systems to crash. This activity could indicate a DoS attack.
- Fragments overwrite data—Some early fragments in a sequence are transmitted along with random data. Later fragments overwrite the random data. If the packet is not reassembled properly, the IDPS cannot detect the attack.
- Fragments are too small—If any fragment other than the final fragment in a sequence is less than 400 bytes, it has probably been crafted intentionally. Such a small fragment is probably part of a DoS attack.

### IPv6

- Fragments with a destination address of a network device—Assembly of fragmented packets occurs only at the destination host. If a router, firewall, or other device is the destination of fragmented IPv6 packets, a denial of service attack might be intended.
- Fragments are too small—If any fragment other than the final fragment in a sequence is less than 1280 bytes, it has probably been crafted intentionally.

- Fragments that arrive too slowly—Fragments that take more than 60 seconds to deliver should be dropped because they are probably part of an effort to avoid detection.

## Advanced Attacks

Most types of attacks discussed so far have been protocol anomalies—violations of the protocol rules described in RFC statements. Some especially complex attacks use path names, hexadecimal codes, and obfuscated directory names to fool an IDPS into letting the packet through without triggering an alarm. Some of the more advanced IDPS evasion techniques include the following:

- Polymorphic buffer overflow attacks—These attacks are as complicated as they sound. A tool called ADMmutate is used to alter an attack's shell code so that it differs slightly from the known signatures many IDPSs use. After the attacking packets elude the IDPS and reach their intended target, they reassemble into their original form.
- Path obfuscation—A directory path statement in the payload of a packet is obfuscated by using multiple forward slashes. For example, `/Windows/. /. /. /` is essentially the same as `/Windows`. However, because the signatures do not match exactly, an IDPS might be unable to detect this attack.
- CGI scripts—A series of packets is sent to a series of well-known Common Gateway Interface (CGI) scripts, which are scripts used to process data submitted over the Internet. Examples include CGI scripts such as `Count.cgi`, `FormMail`, `AnyForm`, `Php.cgi`, `TextCounter`, and `GuestBook`. You can be certain that someone is attempting to exploit your network if it does not contain these files, but packets attempt to locate them anyway.
- Packet injection—With readily available tools like Nemesis, attackers can easily craft packets that comply with protocols like ARM, DNS, ICMP, Ethernet, IGMP, IP, OSPF, RIP, TCP, and UDP. These packets can be inserted into network traffic. While these tools are useful for testing IDPSs and firewalls, they can also be used to disrupt communications, spoof a variety of systems, and carry out a number of attacks.

The only way to avoid these attacks is to keep your IDPS signatures up to date and to watch your log files closely.

## Remote Procedure Call Attacks

Remote Procedure Calls (RPC) is a standard set of communications rules that allows one computer to request a service (in other words, a remote procedure) from another computer on a network. RPC uses a program called a portmapper that maintains a record of each remotely accessible program and the port it uses. The RPC portmapper is actually a service that runs on the system and converts RPC program numbers into TCP/IP protocol port numbers. Because RPC can provide remote access to applications, attackers naturally attempt to use it to gain unauthorized access to those applications. Here are some examples of RPC-related events that should trigger IDPS alarms:

- RPC dump—A targeted host receives an RPC dump request: a request to report the presence and port use of any RPC services the system provides.
- RPC set spoof—A targeted host receives an RPC set request from a source IP address of 127.0.0.1.
- RPC NFS sweep—A targeted host receives a series of requests for the NFS program on a succession of different ports.

RPC services such as Network Information System (NIS) use a four-byte service number because there are too many services to use a two-byte port number. When an RPC service starts, it allocates a random TCP or UDP port for itself. It then contacts rpcbind or portmapper and registers its service number and TCP/UDP port. Portmapper and rpcbind always run on port 111, for example. A client that wants to talk to a server first contacts portmapper to get the port number, and then continues the exchange with the server directly. A client can bypass portmapper and scan for services. There is no guarantee that a particular service will end up on a particular port.

◀ Preliminary Activity for Week 7

Jump to...



Analysis, Application, and Exploration for Week 7 ▶



## Navigation

Home



Dashboard

Site pages

My courses

Capstone Project 2

Network Defense and Remote Access Configuration

Participants


General

O6 - Preliminary Examination

O7 [Enter Module Title Here]

 Preliminary Activity for Week 7

 **Lesson Proper for Week 7**

 Analysis, Application, and Exploration for Week 7

 Generalization for Week 7

 Evaluation for Week 7

 Assignment for Week 7

O8 [Enter Module Title Here]

OJT/Practicum 2

Seminars and Tours

Courses



## 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

---

Bestlink College of the Philippines  
College Department

Powered by [eLearning Commons](#)