(2½ Hours)

[Total Marks: 60]

N. B.: (1) **All** questions are **compulsory**.
(2) Make **suitable assumptions** wherever necessary and **state the assumptions** made.
(3) Answers to the **same question** must be **written together**.
(4) Numbers to the **right** indicate **marks**.
(5) Draw **neat labelled diagrams** wherever **necessary**.
(6) Use of **Non-programmable** calculators is **allowed**.

I	**Choose the correct alternative and rewrite the entire sentence with the correct alternative. (30)**			
1.	A ____________ can work in two ways: synchronously and asynchronously.			
	a.	Token device	**b.**	Digital Transmitter
	c.	password	**d.**	Signal Jammer
2.	Which amongst these is not an attribute specified in the X.500 Schema			
	a.	Country	**b.**	Organization
	c.	Username	**d.**	Location
3.	NIST SP 800-124 proposes a five-phase lifecycle model for an enterprise mobile device solution. Which amongst this is not one of them.			
	a.	Initiation	**b.**	Development
	c.	Disposal	**d.**	Configuration
4.	Syslog uses which ports?			
	a.	8080	**b.**	514
	c.	443	**d.**	540
5.	______________ was created to provide a standardized solution for security automation.			
	a.	SCAP	**b.**	SAML
	c.	MDM	**d.**	VA/PT
6.	A _____________ is a proxy server that resides in front of the application servers (normally web servers) and functions as an entry point for Internet users who want to access the corporate internal web application resources.			
	a.	Firewall Proxy	**b.**	Forward proxy
	c.	Secure Proxy	**d.**	Reverse proxy
7.	Processes can be grouped together and managed as a unit called a _____			
	a.	thread	**b.**	job
	c.	Work	**d.**	pool
8.	Windows uses _____________ to specify the current security context for a process.			
	a.	Tokens	**b.**	Workers

	c.	Threads	**d.**	Privileges
9.	A __________ process is a process that releases its associated memory and resources but remains in the entry table.			
	a.	dead	**b.**	thread
	c.	zombie	**d.**	orphan
10.	File permissions for a file or directory can be modified using the _________ command,			
	a.	chper	**b.**	chassoc
	c.	chdir	**d.**	chmod
11.	__________ limits the impact of security vulnerabilities and bugs in code to only run inside.			
	a.	Sandboxing	**b.**	Emulation
	c.	Simulation	**d.**	Virtualization
12.	Which is a piece of malware or configuration change that allows attackers to control the victim’s system remotely.			
	a.	Botnet	**b.**	Backdoor
	c.	Ransomware	**d.**	Rootkit
13.	________ a segment of the entire pathway that an attack uses to access a vulnerability.			
	a.	Attack vector	**b.**	Privilege Escalation
	c.	Threat vector	**d.**	Vulnerability Exploitation
14.	What ensures that data that is maintained is accurate, complete, and protected from unauthorized modification			
	a.	Encryption	**b.**	Integrity
	c.	Non-Repudiation	**d.**	Passwords
15.	__________ is the way you document and preserve evidence from the time that you started the cyber forensics investigation to the time the evidence is presented in court.			
	a.	Static Safe Storage	**b.**	Digital Storage
	c.	Chain of custody	**d.**	Write Blocking
16.	A __________ is a unit of execution that is manually scheduled by an application.			
	a.	Thread	**b.**	Process
	c.	Job	**d.**	Fiber
17.	The __________ command is used with the regular expression in linux while searching files.			
	a.	grep	**b.**	Cat
	c.	find	**d.**	ls
18.	Which amongst these is a security artefact			
	a.	Hashes	**b.**	Hostname

	c.	URI/URL	**d.**	All of Above
19.	VERIS stands for ________________			
	a.	Vocabulary of Event Recording and Incident Sharing	**b.**	Vocabulary of Event Registration and Incident Sharing
	c.	Vocabulary of Event Registration and Incident Stopping	**d.**	Vocabulary of Event Remediation and Incident Suspension
20.	One of the most widely adopted standards to calculate the severity of a given vulnerability is____________________			
	a.	SAML	**b.**	CVSS
	c.	VERIS	**d.**	CSIR
21.	______________ is a repeatable and measurable process we've designed to increase the resiliency and trustworthiness of our products			
	a.	PDM	**b.**	SAML
	c.	SDL	**d.**	VERIS
22.	The purpose of ________ is forcing any publicly held company to have internal controls and procedures for financial reporting to avoid future corporate fraud			
	a.	HIPPA	**b.**	PCI DSS
	c.	SOX	**d.**	PDPA
23.	________ involves identifying something based on various characteristics specified in a detector			
	a.	Enumeration	**b.**	Cloning
	c.	Assessment	**d.**	Profiling
24.	_______ can be used to prioritize traffic to guarantee performance of specific traffic types such as voice and video			
	a.	QoS	**b.**	Fiber optics
	c.	Bandwidth	**d.**	Speed
25.	_____________ is the process of capturing, storing, and analyzing data so that it exists in only one form.			
	a.	Regularization	**b.**	Standardization
	c.	Normalization	**d.**	Trimming
26.	Which amongst these is not a 5 tuple parameter			
	a.	Source Port	**b.**	Protocol
	c.	Destination Address	**d.**	MAC Address
27.	Which amongst these is a metadata tool for finding unintended data about the target system.			
	a.	DuckDuckGo	**b.**	Shodan
	c.	VirusTotal	**d.**	ExploitDB

28.	________________ is research on a target and typically is the most time-consuming yet most rewarding step of the kill chain			
	a.	Enumeration	**b.**	Exploitation
	c.	Reconnaissance	**d.**	Sniffing
29.	The attacker uses __________ technique to direct a customer’s URL from a valid resource to a malicious one that could be made to appear as the valid site to the user.			
	a.	Phishing	**b.**	Pharming
	c.	Spoofing	**d.**	Snooping
30.	_____________ is a type of web application vulnerability where malicious scripts are injected into legitimate and trusted websites.			
	a.	XSS	**b.**	RCE
	c.	SQL Injection	**d.**	DNS Poisoning

II	**Attempt *any one* of the following:**		**6**
	a)	Explain configuration and Change Management and discuss the ITIL Service Transition steps in detail.	

A NIST SP 800-128 defines configuration management as a set of activities used to maintain organizational resource integrity through the control of processes for initializing, changing, and monitoring the resource configuration. A configuration item (CI) is defined as an identifiable part of the system that is the target of the configuration control process. A CI can be an information system component such as a router, application, server, or a group of components (for example, a group of routers sharing the same operating system and configuration), or it can be a noncomponent such as documentation or firmware. Each CI includes a set of attributes; for example, the attributes for a CI describing a server could be the firmware version and applications installed. If these attributes are configured as individual CIs, then two CIs are said to bem“in relation.” For example, a Cisco router could be considered a CI, and the router operating system, IOS-XE 16.1.1, could be considered a separate CI. These two CIs are said to be “in relation.”

The set of attributes and relationships for a CI create a configuration record. The configuration record is stored in the configuration management database (CMDB). The main goal of configuration management is to manage the lifecycle of the CIs. An important step is the inventory of CIs. The inventory process is about identifying all the CIs and capturing the configuration records in the configuration management database.

Another important concept in configuration management is the baseline configuration. A baseline configuration is a set of attributes and CIs related to a system, which has been formally reviewed and approved. It can only be changed with a formal change process.

While configuration management goes beyond information security, it is an important part of the management of secure configurations, as well as to enable security and facilitate the assessment of the risk for an organization.

Change Management

A change is defined as any modification, addition, or removal of an organizational resource (for example, of a configuration item). Change management includes all policies, processes, and technologies for handling a change’s lifecycle.

In ITIL Service Transition, changes are categorized as follows:

		• **Standard change**: A common change that has already been authorized and is low risk. This type of change might not need to follow a formal change management process. • **Emergency change**: A change that needs to be implemented on an urgent basis. This type of change usually has a separate procedure. • **Normal change**: A change that is not a standard change or an emergency change. This is the type of change that will go through the full change management procedure. According to ITIL Service Transition, a change control process includes the following steps: **Step 1.** Create an RFC. In this step, an RFC is created with a high-level plan for the change and its motivation. **Step 2.** Record the RFC. In this step, the RFC is formally recorded in the change management system. **Step 3.** Review the RFC. In this step, the RFC is reviewed to see whether the change makes sense and whether it is necessary to proceed further in the process. **Step 4.** Assess and evaluate the change. In this step, the change review board will determine whether the change requires change control (for example, if it was already preapproved). In this step, the security impact of the change is also determined. **Step 5.** Authorize the change's build and test. The change authority is appointed and the change test plan is formally authorized. The test may be built before the actual authorization and authorization decision is taken based on the outcome of the test. The test should confirm the security impact anticipated in step 4 or highlight additional impacts. **Step 6.** Coordinate the change's build and test. The authorized change is passed to the technical group for the change's build and testing. **Step 7**. Authorize deployment. If the change's build and testing phase goes fine, the change is authorized for deployment. The change authority may request additional tests and send the change back to previous steps. **Step 8**. Implement the change. The change is implemented. **Step 9.** Review and close the change record.. After the change is deployed, the system is tested to make sure the change was deployed correctly. If all goes well, the change record is updated in the change management system and the request is closed.	
	b)	Discuss CVSS base, temporal, and environmental groups in detail.	
	A	The CVSS score is obtained by taking into account the base, temporal, and environmental group information. The score for the base group is between 0 and 10, where 0 is the least severe and 10 is assigned to highly critical vulnerabilities (for example, for vulnerabilities that could allow an attacker to remotely compromise a system and get full control). Additionally, the score comes in the form of a vector string that identifies each of the components used to make up the score. The formula used to obtain the score takes into account various characteristics of the vulnerability and how the attacker is able to leverage these characteristics. At press time, the latest version of the CVSS framework is version 3 (CVSSv3). CVSSv3 defines several characteristics for the base, temporal, and environmental groups. The base group defines exploitability metrics that measure how the vulnerability can be exploited, and impact metrics that measure the impact on confidentiality, integrity, and availability. In addition to these two, a metric called scope change (S) is used to convey the impact on systems that are affected by the vulnerability but do not contain vulnerable code. Exploitability metrics include the following:	

Attack Vector (AV): Represents the level of access an attacker needs to have to exploit a vulnerability. It can assume four values:

- Network (N)
- Adjacent (A)
- Local (L)
- Physical (P)

Attack Complexity (AC): Represents the conditions beyond the attacker's control that must exist in order to exploit the vulnerability. The values can be one of the following:

- Low (L)
- High (H)

Privileges Required (PR): Represents the level of privileges an attacker must have to exploit the vulnerability. The values are as follows:

- None (N)
- Low (L)
- High (H)

User Interaction (UI): Captures whether user interaction is needed to perform an attack. The values are as follows:

- None (N)
- Required (R)

Scope (S): Captures the impact on other systems other than the system being scored. The values are as follows:

- Unchanged (U)
- Changed (C)

The Impact metrics include the following:

Confidentiality Impact (C): Measures the degree of impact to the confidentiality of the system. It can assume the following values:

- Low (L)
- Medium (M)
- High (H)

Integrity Impact (I): Measures the degree of impact to the integrity of the system. It can assume the following values:

- Low (L)
- Medium (M)
- High (H)

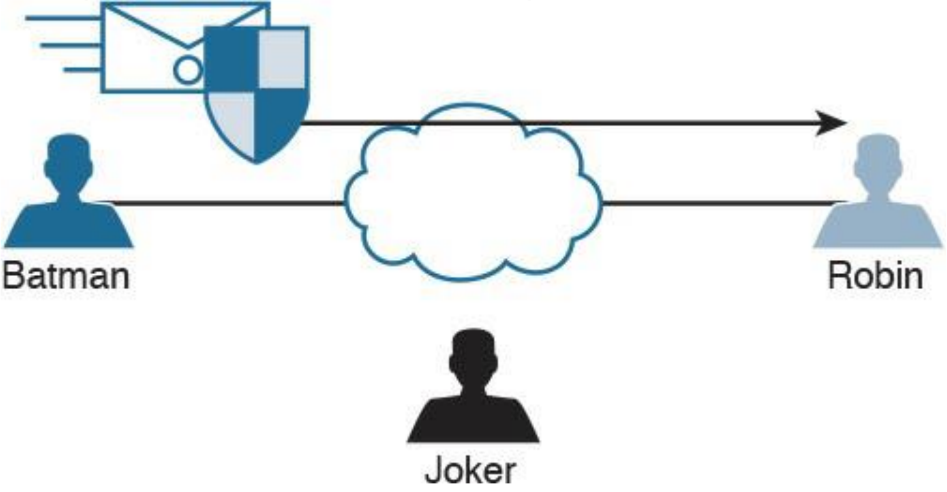
Availability Impact (A): Measures the degree of impact to the availability of the system. It can assume the following values:

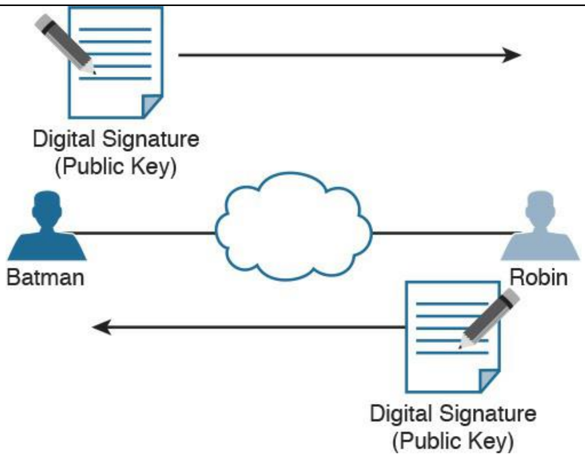
- Low (L)
- Medium (M)
- High (H)

The temporal group includes three metrics:

- **Exploit code maturity (E):** Measures whether or not public exploits are available.
- **Remediation Level (RL):** Indicates whether a fix or workaround is available.
- **Report Confidence (RC):** Indicates the degree of confidence in the existence of the vulnerability.

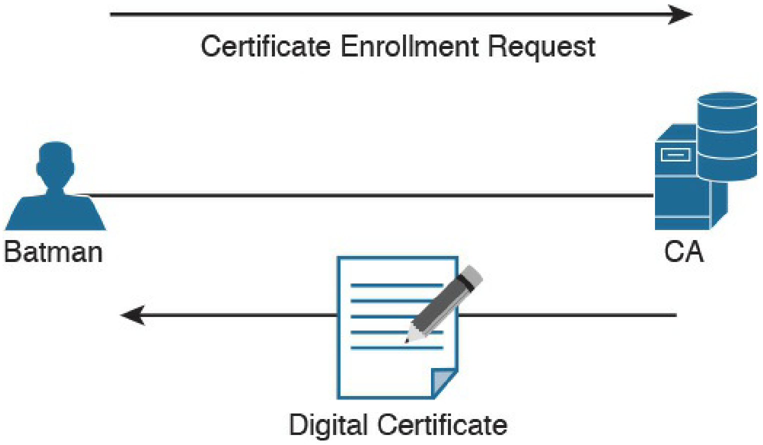
The environmental group includes two main metrics:

		• **Security Requirements (CR, IR, AR):** Indicate the importance of confidentiality, integrity, and availability requirements for the system. • **Modified Base Metrics (MAV, MAC, MAPR, MUI, MS, MC, MI, MA):** Allow the organization to tweak the base metrics based on specific characteristics of the environment.	
	c)	Discuss and explain the working of Digital signatures with an example.	
	A	When you sign something, this often represents a commitment to follow through, or at least proves that you are who you say you are. In the world of cryptography, a digital signature provides three core benefits: • Authentication • Data integrity • Nonrepudiation One of the best ways to understand how a digital signature operates is to remember what you learned in the previous sections about public and private key pairs, hashing, and encryption. Digital signatures involve each of these elements. In most security books, three fictional characters are used to explain encryption and PKI: Bob, Alice, and Eve. Bob and Alice typically are the two entities that exchange a secured message over a public or untrusted network, and Eve is the person who tries to "eavesdrop" and steal the information being exchanged. In this book, let's make it more entertaining and use Batman, Robin, and the Joker. In Figure 6-1, all three entities are illustrated. Batman wants to send an encrypted message to Robin without the Joker being able to read it.  Batman and Robin are two people who want to establish a VPN connection to each other, and to do so they want to use digital signatures to verify each other to make sure they are talking to the right entity.	



Both Batman and Robin want to verify each other, but for simplicity let's focus on one entity: Batman wanting to prove its identity to the other device, Robin. (This could also be phrased as Robin asking Batman to prove Batman's identity.)

As a little setup beforehand, you should know that both Batman and Robin have generated public-private key pairs, and they both have been given digital certificates from a common certificate authority (CA). A CA is a trusted entity that hands out digital certificates.



In Figure, Batman requests a digital certification from (enrolls with) a CA, and the CA assigns one to Batman. If you and I were to open the digital certificate, we would find the name of the entity (in this case, Batman). We would also find Batman's public key (which Batman gave to the CA when applying for the digital certificate). Figure shows an example of a digital certificate. In this case, Cisco's website (cisco.com) digital certificate is shown. Also, the digital signature of the CA is shown.

Both Batman and Robin trust the CA and have received their certificates.

Batman takes a packet and generates a hash. Batman then takes this small hash and encrypts it using Batman's private key. (Think of this as a shipping container, and Batman is using the small key in the small keyhole to lock the data.) Batman attaches this encrypted hash to the packet and sends it to Robin. The fancy name for this encrypted hash is digital signature . When Robin receives this packet, it looks at the encrypted hash that was sent and decrypts it using Batman's public key. (Think of this as a big keyhole and the big key being used to unlock the data.) Robin then sets the decrypted hash off to the side for one moment and runs the same hash algorithm on the packet it just received. If the hash Robin just calculated matches the hash just received (after Robin decrypted it using the sender's public key), then Robin knows two things: that the only person who could have encrypted it was Batman with Batman's private key, and that the data

		integrity on the packet is solid, because if one bit had been changed, the hashes would not have matched. This process is called authentication, using digital signatures, and it normally happens in both directions with an IPsec VPN tunnel if the peers are using digital signatures for authentication (referred to as rsa-signatures in the configuration). At this point you might be wondering how Robin got Batman’s key (Batman’s public key) to begin with. The answer is that Batman and Robin also exchanged digital certificates that contained each other’s public keys. Batman and Robin do not trust just any certificates, but they do trust certificates that are digitally signed by a CA they trust. This also implies that to verify digital signatures from the CA, both Batman and Robin also need the CA’s public key. Most browsers and operating systems today have the built-in certificates and public keys for the mainstream CAs on the Internet. Figure 6-5 shows the “System Roots” keychain on Mac OS X.	
2	**Attempt *<u>any one</u>* of the following:**		**6**
	a)	What is Windows Management Instrumentation? Explain in Detail.	
	A	The next topic focuses on managing Windows systems and sharing data with other management systems. Windows Management Instrumentation (WMI) is a scalable system management infrastructure built around a single, consistent, standards-based, extensible, object-oriented interface. Basically, WMI is Microsoft’s approach to implementing Web-Based Enterprise Management (WBEM), which is a tool used by system management application developers for manipulating system management information. WMI uses the Common Information Model (CIM) industry standard to represent systems, applications, networks, devices, and other managed components. CIM is developed and maintained by the Distributed Management Task Force (DMTF). It is important to remember that WMI is only for computers running Microsoft Windows. WMI comes preinstalled on all computers running Windows Millennium Edition (ME), Windows 2000, Windows XP, or Windows Server 2003; however, it can be downloaded to older systems running Windows 95, Windows 98, or Windows NT 4.0. The purpose of WMI is to define a set of proprietary environment-independent specifications used for management information that’s shared between management applications. WMI allows scripting languages to locally and remotely manage Microsoft Windows computers and services. The following list provides examples of what WMI can be used for: • Providing information about the status of local or remote computer systems • Configuring security settings • Modifying system properties • Changing permissions for authorized users and user groups • Assigning and changing drive labels • Scheduling times for processes to run • Backing up the object repository • Enabling or disabling error logging Using WMI by itself doesn’t provide these capabilities or display any data. You must pull this information using scripts and other tools. WMI can be compared to the electronics data of a car, where the car dashboard is the tool used to display what the electronics are doing. Without the dashboard, the electronics are there, but you won’t be able to interact with the car or obtain any useful data. An example of WMI would be using a script to display the time zone configured on a	

Windows computer or issuing a command to change the time zone on one or more Windows computers.

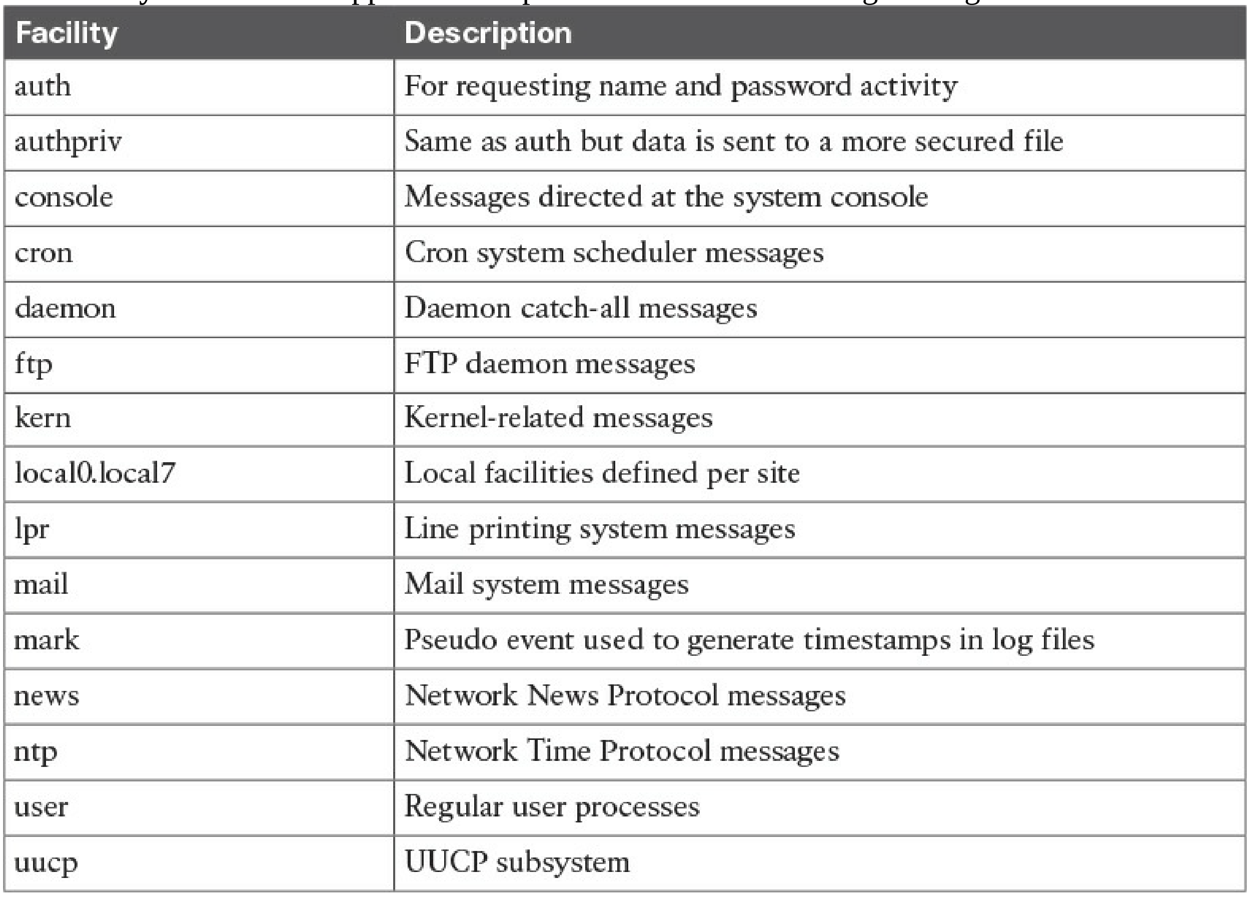
When considering Windows security, you should note that WMI could be used to
perform malicious activity. Malicious code could pull sensitive data from a system or automate malicious tasks. An example would be using WMI to escalate privileges so that malware can function at a higher privilege level if the security settings are modified. Another attack would be using WMI to obtain sensitive system information.

There haven’t been many WMI attacks seen in the wild; however, Trend Micro
published a whitepaper on one piece of WMI malware called TROJ_WMIGHOST.A. So although such attacks are not common, they are possible. WMI requires administrative permission and rights to be installed; therefore, a best practice to protect systems against this form of exploitation is to restrict access to the WMI service.

b) Discuss Unix bases syslog in brief. Explain different components of syslog.

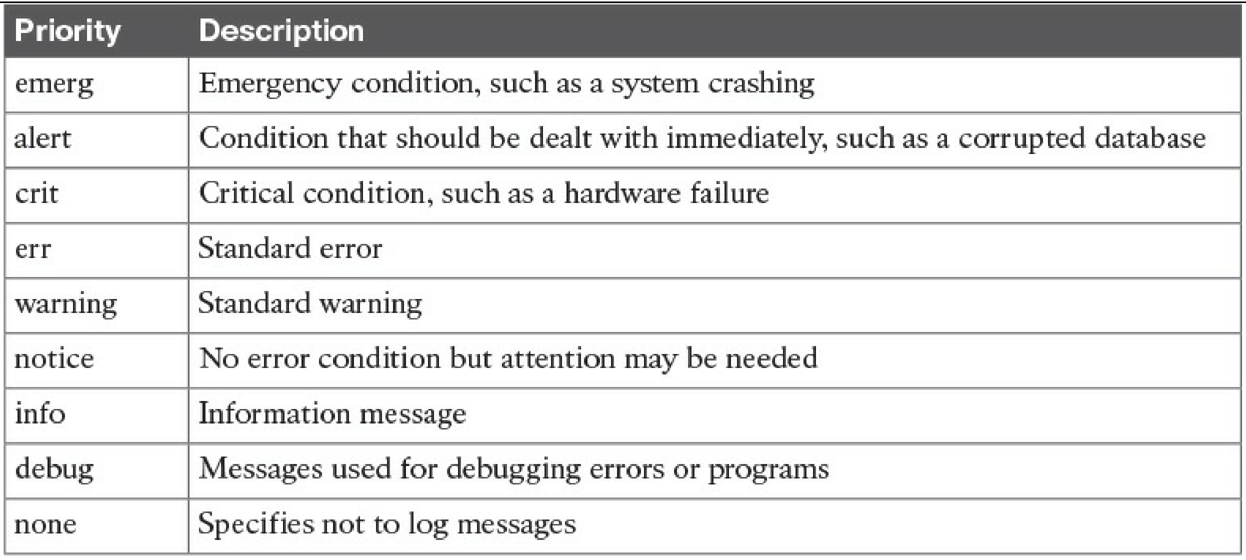
A UNIX-based systems have very flexible logging capabilities, enabling the user to record just about anything. The most common form of logging is the general-purpose logging facility called syslog. Most programs send logging information to syslog. Syslog is typically a daemon found under the /var/log directory. You can see the logs by typing cd /var/log followed by ls to view all the logs. Make sure you know the location of these files.

The **facility** describes the application or process that submits the log message.



Facility	Description
auth	For requesting name and password activity
authpriv	Same as auth but data is sent to a more secured file
console	Messages directed at the system console
cron	Cron system scheduler messages
daemon	Daemon catch-all messages
ftp	FTP daemon messages
kern	Kernel-related messages
local0.local7	Local facilities defined per site
lpr	Line printing system messages
mail	Mail system messages
mark	Pseudo event used to generate timestamps in log files
news	Network News Protocol messages
ntp	Network Time Protocol messages
user	Regular user processes
uucp	UUCP subsystem

All messages are not treated the same. A **priority** is used to indicate the level of importance of a message.



Priority	Description
emerg	Emergency condition, such as a system crashing
alert	Condition that should be dealt with immediately, such as a corrupted database
crit	Critical condition, such as a hardware failure
err	Standard error
warning	Standard warning
notice	No error condition but attention may be needed
info	Information message
debug	Messages used for debugging errors or programs
none	Specifies not to log messages

Transaction logs record all transactions that occur. For example, a database transaction log would log any modifications to the database. **Alert logs** record errors such as a startup, shutdown, space errors, and so on. **Session logs** track changes made on managed hosts during a web-based system manager session. Logging occurs each time an administrator uses web-based system management to make a change on a host. **Threat logs** trigger when an action matches one of the security profiles attached to a security rule. It is important to distinguish what type of log would go where for an event scenario. An example would be knowing that a system crash would be an alert log and that a malicious attack would be a threat log. Actions such as logging are triggered by selectors.

Selectors monitor for one or more facility and level combinations and, when triggered, perform some action. When a specific priority level is specified, the system will track everything at that level as well as anything higher. For example, if you use crit, you will see messages associated with crit, alert, and emerg. This is why enabling debug is extremely chatty because you are essentially seeing all messages.

Actions are the results from a selector triggering on a match. Actions can write to the log file, echo the message to the console or to other devices so users can read it, send a message to another syslog server, and perform other actions.

c) Explain Host-Based Firewalls and Host-Based Intrusion Prevention in detail.

A Host-based firewalls are often referred to as “personal firewalls.” Personal firewalls and host intrusion prevention systems (HIPSs) are software applications that you can install on end-user machines or servers to protect them from external security threats and intrusions. The term personal firewall typically applies to basic software that can control Layer 3 and Layer 4 access to client machines. HIPS provides several features that offer more robust security than a traditional personal firewall, such as host intrusion prevention and protection against spyware, viruses, worms, Trojans, and other types of malware.

Today, more sophisticated software is available on the market that makes basic personal firewalls and HIPS obsolete. For example, Cisco Advanced Malware Protection (AMP) for Endpoints provides more granular visibility and controls to stop advanced threats missed by other security layers. Cisco AMP for Endpoints takes advantage of telemetry from big data, continuous analysis, and advanced analytics provided by Cisco threat intelligence in order to detect, analyze, and stop advanced malware across endpoints.

Cisco AMP for Endpoints provides advanced malware protection for many operating systems, including the following:

- Windows
- Mac OS X
- Android

Attacks are getting very sophisticated, and they can evade detection of traditional systems and endpoint protection. Nowadays, attackers have the resources, knowledge, and persistence to beat point-in-time detection. Cisco AMP for Endpoints provides mitigation capabilities that go beyond point-in-time detection. It uses threat intelligence from Cisco to perform retrospective analysis and protection. Cisco AMP for Endpoints also provides device and file trajectory capabilities to allow the security administrator to analyze the full spectrum of an attack.

Cisco acquired a security company called Threat Grid that provides cloud-based and on-premises malware analysis solutions. Cisco integrated Cisco AMP and Threat Grid to provide a solution for advanced malware analysis with deep threat analytics. The Cisco AMP Threat Grid integrated solution analyzes millions of files and correlates them against hundreds of millions of malware samples. This provides a lot of visibility into attack campaigns and how malware is distributed. This solution provides security administrators with detailed reports of indicators of compromise and threat scores that help them prioritize mitigations and recovery from attacks.

In addition to host-based firewalls and HIPS, there are several solutions that provide hardware and software encryption of endpoint data. Several solutions provide capabilities to encrypt user data "at rest," and others provide encryption when transferring files to the corporate network.

When people refer to email encryption, they often are referring to encrypting the actual email message so that only the intended receiver can decrypt and read the message. To effectively protect your emails, however, you should make sure of the following:

- The connection to your email provider or email server is actually encrypted.
- Your actual email messages are encrypted.
- Your stored, cached, or archived email messages are also protected.

There are many commercial and free email encryption software programs. The following are examples of email encryption solutions:

- Pretty Good Privacy (PGP)
- GNU Privacy Guard (GnuPG)
- Secure/Multipurpose Internet Mail Extensions (S/MIME)
- Web-based encryption email services such as Sendinc and JumbleMe

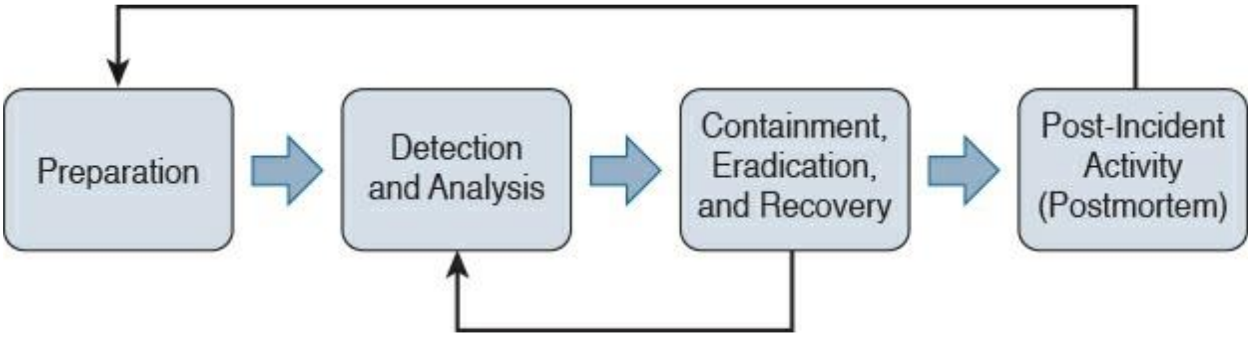
S/MIME requires you to install a security certificate on your computer, and PGP requires you to generate a public and private key. Both require you to give your contacts your public key before they can send you an encrypted message. Similarly, the intended recipients of your encrypted email must install a security certificate on their workstation or mobile device and provide you with their public key before they send the encrypted email (so that you can decrypt it). Many email clients and web browser extensions for services such as Gmail provide support for S/MIME. You can obtain a certificate from a certificate authority in your organization or from a commercial service such as DigiCert or VeriSign. You can also obtain a free email certificate from an organization such as Comodo.

Many commercial and free pieces of software are available that enable you to encrypt files in an end-user workstation or mobile device. The following are a few examples of free solutions:

- **GPG:** GPG enables you to encrypt files and folders on a Windows, Mac, or Linux system. The built-in Mac OS X Disk Utility: Disk Utility enables you to create secure disks by encrypting files with AES 128-bit or AES 256-bit encryption.
- **TrueCrypt:** An encryption tool for Windows, Mac, and Linux systems.
- **AxCrypt**: A Windows-only file encryption tool.

		• **BitLocker**: A full disk encryption feature included in several Windows operating systems. Many Linux distributions such as Ubuntu: Allow you to encrypt the home directory of a user with built-in utilities. • **Mac OS X FileVault**: Supports full disk encryption on Mac OS X systems. The following are a few examples of commercial file encryption software: • Symantec Endpoint Encryption • PGP Whole Disk Encryption • McAfee Endpoint Encryption (SafeBoot) • Trend Micro Endpoint Encryption	
3	**Attempt *<u>any one</u>* of the following:**		**6**
	a)	What is threat modelling? Why is it Important? Explain any one Threat Modelling Framework in Detail.	
	A	Risk analysis is crucial. You need to know what you are protecting and how you are protecting it. What are your critical systems and assets? What constitutes your organization today? These are some initial questions you should ask yourself when starting any risk analysis process. You must know the difference between threats and vulnerabilities. Threats are occurrences that can affect a system or an organization as a whole. Examples of threats include fraud, theft of information, and physical theft. Vulnerabilities are flaws that make a system, an individual, or an organization exposed and susceptible to a threat or an attack. Typically, when you ask security engineers, managers, architects, and executives to list or describe the critical systems of their organization, their answers are contradictory. One of the main goals that members of an organization should have is to understand their environment to better comprehend what they are trying to protect and what risks are most imminent. Several methods of risk analysis have been published in books, websites, magazines, and blogs. Some take the quantitative approach, some take the qualitative approach, and others measure impact versus probability. The primary goal of any threat modeling technique is to develop a formal process while identifying, documenting, and mitigating security threats. This process has a huge impact on any organization because it is basically a methodology used to understand how attacks can take place and how they will impact the network, systems, and users. Organizations have adopted several threat modeling techniques. For example, Microsoft uses the DREAD model. The DREAD acronym defines five key areas: • Damage potential • Reproducibility • Exploitability • Affected users • Discoverability In the DREAD model, the first step is to quantify or estimate the damage potential of a specific threat. This estimate can include monetary and productivity costs, followed by a probability study on the reproducibility and exploitability of the vulnerability at hand. In addition, the first step should identify which users and systems will be affected and how easily the threat can be discovered and identified. spoofing, tampering, repudiation, information disclosure, denial of service, and elevation of privilege. STRIDE was created by Loren Kohnfelder and Praerit Garg. This is a framework designed to help software developers identify the types of threats against the applications they are creating.	

		The Spoofing: Sometimes referred to as identify spoofing. Attackers can disguise themselves as someone else. They can also disguise their systems as some other systems. For instance, in many distributed denial-of-service (DDoS) attacks, attackers can spoof the source of the attacks (that is, the IP addresses of the attacking machines or bots) in order to carry out the attack and maintain anonymity. This is why systems should have protection in place against spoofing attacks—and not just for DDoS. In general, users should not be able to become any other users or assume the attributes of other users, period. **Tampering:** This ties into the discussion earlier in this chapter about integrity. Users must not be able to tamper with data, applications, or systems. In threat modeling, you must understand what threats could allow an attacker to tamper with data, applications, or systems in your organization. **Repudiation**: You must consider if the system or applications requires nonrepudiation controls, such as system logs, web access logs, and audit trails. Another consideration is that an application should run with the user's privileges, not more. **Information disclosure:** You must make sure that a system or application does not disclose information that is not intended. For example, a web application should not store usernames and passwords in its source. Also, user credentials should not be stored in logs or in any other configuration or troubleshooting feature in plain text. **Denial of service:** You should evaluate what threats can cause a denial-of- service condition. This is beyond just performance testing and should employ methodologies such as fuzzing (sending random data to an following are the different components of STRIDE: application or protocol). **Elevation of privilege:** It is very important that you ensure in any application or system that users cannot elevate their privileges. Many organizations develop an authorization matrix to ensure that only authorized users and roles can access privileged functionality. Another threat modeling technique is to create attack trees. Bruce Schneier, the chief technology officer of Counterpane Internet Security and the inventor of the Blowfish and Twofish encryption algorithms, initially introduced this method. Attack trees represent attacks against a system or network in a hierarchical tree structure. The root node describes a goal, and the leaf nodes are various ways of reaching such a goal. For example, the main goal of a specific attack may be to interrupt the services of an ecommerce web server farm. This goal will be the root of the tree. Each subsequent "tree branch or leaf" describes the methods used to take down that web server farm (such as sending millions of spoofed TCP packets, compromising zombies on the Internet to launch DDoS attacks, and so on).	
	b)	What is the Role of Attribution in a Cybersecurity Investigation Explain different types of digital evidence in detail with examples.	
	A	One of the key topics in cybersecurity forensics is attribution of assets and threat actors. There is undeniable motivation to support an evidence-led approach to cybersecurity forensics to achieve good attribution. A suspect-led approach is pejorative and often biased to the disadvantage of those being investigated. Due to the large number of technical complexities, it is often impractical for cybersecurity forensics experts to determine fully the reliability of endpoints, servers, or network infrastructure devices and provide assurances to the court about the soundness of the processes involved and the complete attribution to a threat actor. The forensics expert needs to ensure that not one part of the examination process is overlooked or repetitive. In addition, cybersecurity forensic experts are often confronted with the inefficacy of traditional security processes in systems and networks designed to preserve documents and network functionality—especially because most systems are not designed to enhance digital evidence recovery. There is a need for appropriate cybersecurity forensic tools, including software	

		imaging and the indexing of increasingly large datasets in order to successfully reconstruct an attack and attribute the attack to an asset or threat actor. One thing to keep in mind is that traditional digital forensics tools are typically designed to obtain the “lowest-hanging fruit” and encourage security professionals to look for the evidence that is easiest to identify and recover. Often, these tools do not have the capability to even recognize other, less-obvious evidence.	
	c)	Discuss False Positives, False Negatives, True Positives, and True Negatives in Detail with Examples.	
	A	The term false positive is a broad term that describes a situation in which a security device triggers an alarm but there is no malicious activity or an actual attack taking place. In other words, false positives are “false alarms,” and they are also called “benign triggers.” False positives are problematic because by triggering unjustified alerts, they diminish the value and urgency of real alerts. If you have too many false positives to investigate, it becomes an operational nightmare and you most definitely will overlook real security events. There are also false negatives, which is the term used to describe a network intrusion device’s inability to detect true security events under certain circumstances—in other words, a malicious activity that is not detected by the security device. A true positive is a successful identification of a security attack or a malicious event. A true negative is when the intrusion detection device identifies an activity as acceptable behavior and the activity is actually acceptable. Traditional IDS and IPS devices need to be tuned to avoid false positives and false negatives. Next-generation IPSs do not need the same level of tuning compared to traditional IPS. Also, you can obtain much deeper reports and functionality, including advanced malware protection and retrospective analysis to see what happened after an attack took place.	
4	**Attempt *any one* of the following:**		**6**
	a)	Discuss in detail the Incident Response Process?	
	A	NIST Special Publication 800-61 goes over the major phases of the incident response process in detail. You should become familiar with that publication, as it provides additional information that will help you succeed in your security operations center (SOC). The important key points are summarized here.  Preparation Detection and Analysis Containment, Eradication, and Recovery Post-Incident Activity (Postmortem) **The Preparation Phase** The preparation phase includes creating and training the incident response team, as well as deploying the necessary tools and resources to successfully investigate and resolve cybersecurity incidents. In this phase, the incident response team creates a set of controls based on the results of risk assessments. The preparation phase also includes the following tasks: • Creating processes for incident handler communications and the facilities that will host the security operation center (SOC) and incident response team Making sure that the	

organization has appropriate incident analysis hardware and software as well as incident mitigation software treating risk assessment capabilities within the organization.
- Making sure the organization has appropriately deployed host security network security, and malware prevention solutions
- Developing user awareness training

The Detection and Analysis Phase

The detection and analysis phase is one of the most challenging phases. While some incidents are easy to detect (for example, a denial-of-service attack), many breaches and attacks are left undetected for weeks or even months. This is why detection may be the most difficult task in incident response. The typical network is full of "blind spots" where anomalous traffic goes undetected. Implementing analytics and correlation tools is critical to eliminating these network blind spots. As a result, the incident response team must react quickly to analyze and validate each incident. This is done by following a predefined process while documenting each step the analyst takes. NIST provides several recommendations for making incident analysis easier and more effective:

- Profile networks and systems
- Understand normal behaviors
- Create a log retention policy
- Perform event correlation
- Maintain and use a knowledge base of information
- Use Internet search engines for research
- Run packet sniffers to collect additional data
- Filter the data
- Seek assistance from others
- Keep all host clocks synchronized
- Know the different types of attacks and attack vectors
- Develop processes and procedures to recognize the signs of an incident
- Understand the sources of precursors and indicators
- Create appropriate incident documentation capabilities and processes
- Create processes to effectively prioritize security incidents
- Create processes to effectively communicate incident information (internal and external communications)

Containment, Eradication, and Recovery

The containment, eradication, and recovery phase includes the following activities:

- Evidence gathering and handling
- Identifying the attacking hosts
- Choosing a containment strategy to effectively contain and eradicate the attack, as well as to successfully recover from it

NIST Special Publication 800-61 also defines the following criteria for determining the appropriate containment, eradication, and recovery strategy:

- The potential damage to and theft of resources
- The need for evidence preservation
- Service availability (for example, network connectivity as well as services provided to external parties)
- Time and resources needed to implement the strategy
- Effectiveness of the strategy (for example, partial containment or full containment)

- Duration of the solution (for example, emergency workaround to be removed in four hours, temporary workaround to be removed in two weeks, or permanent solution)

Post-Incident Activity (Postmortem)

The post-incident activity phase includes lessons learned, how to use collected incident data, and evidence retention. NIST Special Publication 800-61 includes several questions that can be used as guidelines during the lessons learned meeting(s):

- Exactly what happened, and at what times?
- How well did the staff and management perform while dealing with the incident?
- Were the documented procedures followed? Were they adequate?
- What information was needed sooner?
- Were any steps or actions taken that might have inhibited the recovery?
- What would the staff and management do differently the next time a similar incident occurs?
- How could information sharing with other organizations be improved?
- What corrective actions can prevent similar incidents in the future?
- What precursors or indicators should be watched for in the future to detect similar incidents?
- What additional tools or resources are needed to detect, analyze, and mitigate future incidents?

b) Explain the Requirements of PCI DSS.

A Being PCI compliant is mandatory for any merchant, processor, acquirer, issuer, or service provider that handles payment card processing. This includes any entities that store, process, or transmit cardholder or authentication data. In addition, PCI compliance is required for payment operations that are outsourced and for third parties involved with payment card processing. The goal of the PCI Data Security Standard (PCI DSS) program is to protect the customer cardholder data when it's processed, stored, or transmitted. Failure to meet PCI could include large fines and have legal ramifications. My personal feeling is PCI DSS needs to be better enforced. That way, in the future when I ask an audience about credit card theft, not so many people will raise their hands, showing that more people have avoided becoming victims due to a failure in a payment transaction. It is very possible that either the current standards are not being enforced properly across all organization types or there is a gap between what PCI DSS requires to be secure versus what is actually needed to secure PCI-related data.

The data being protected by PCI DSS includes the primary account number (PAN), the account data on the magnetic strip, and the data on the embedded chip. This protection is required during the entire sales process, including the vendor or service provider's obligation to release the data upon completing the transaction. Releasing the data is a good thing because attackers can't access stored sensitive data that should no longer be needed after the sales transaction.

Account data defined by PCI DSS 3.2 is shown in the following lists:

- Cardholder data includes the following:
    - Primary account number (PAN)
    - Cardholder name
    - Expiration date
    - Service code
- Sensitive authentication data includes the following:
    - Full track data (magnetic-strip data or equivalent on a chip)
    - CAV2/CVC2/CVV2/CID
    - PINs/PIN blocks

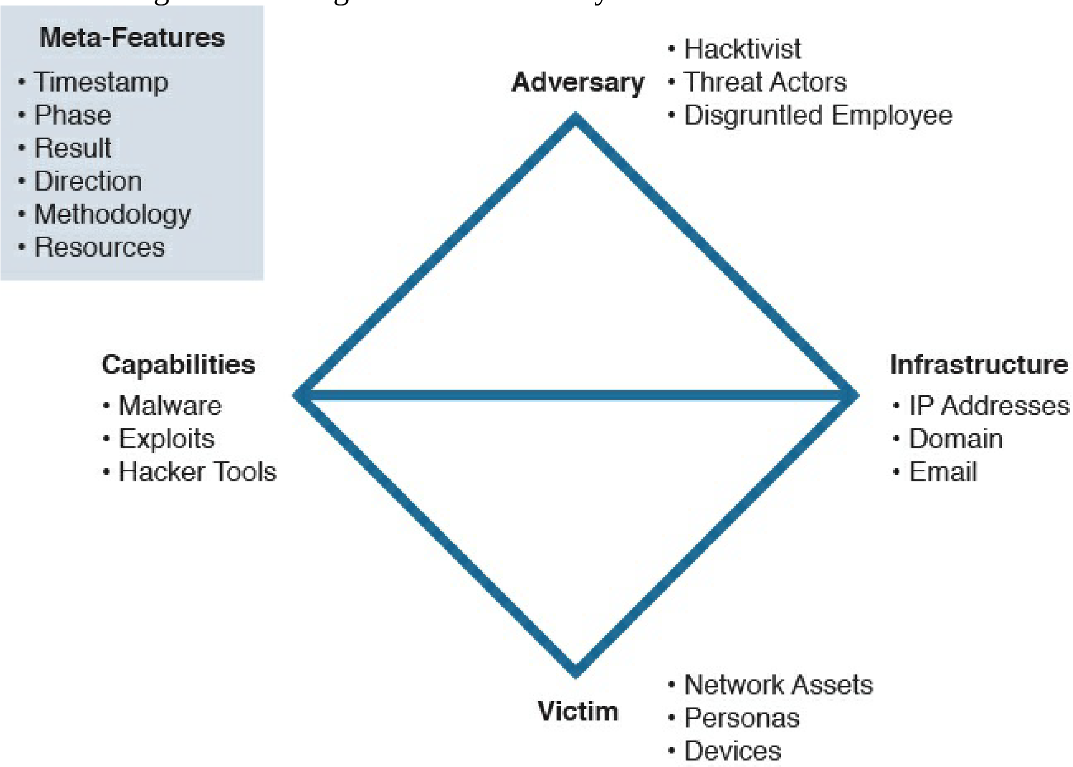
		PCI DSS is very specific about what type of data is and is not permitted to be stored. This includes data that is encrypted, meaning data encryption doesn't give you a reason not to follow PCI DSS standards. It is recommended that you contact the individual payment brand directly to fully understand what data is and is not permitted to be stored. Examples of payment brands are Visa, MasterCard, and American Express. Requirement 1: Build and maintain a secure network and systems. Requirement 2: Do not use vendor-supplied defaults for system passwords and other security parameters. Requirement 3: Protect stored cardholder data. Requirement 4: Encrypt transmission of cardholder data across open, public networks. Requirement 5: Protect all systems against malware and regularly update antivirus software or programs. Requirement 6: Develop and maintain secure systems and applications. Requirement 7: Restrict access to cardholder data by business need to know. Requirement 8: Identify and authenticate access to system components. Requirement 9: Restrict physical access to cardholder data. Requirement 10: Track and monitor all access to network resources and cardholder data. Requirement 11: Regularly test security systems and processes. Requirement 12: Maintain a policy that addresses information security for all personnel.	
	c)	Discuss profiling Network Throughput in Detail.	
	A	Throughput is the amount of traffic that can cross a specific point in the network, and at what speed. If throughput fails, network performance suffers,and most likely people will complain. Administrators typically have alarms monitoring for situations where throughput utilization reaches a level of concern. Normal throughput levels are typically recorded as a network baseline, where a deviation from that baseline could indicate a throughput problem. Baselines are also useful for establishing real throughput compared to what is promised by your service provider. Having a network baseline may sound like a great idea; however, the challenge is establishing what you organization's real network baseline should be. The first step in establishing a network baseline is identifying the devices on your network that can detect network traffic, such as routers and switches. Once you have an inventory, the next step is identifying what type of data can be pulled from those devices. The most common desired data is network utilization; however, that alone should not be your only source for building a network baseline. Network utilization has limitations, such as knowing what devices are actually doing on the network.There are two common tactics for collecting network data for traffic analytic purposes. The first approach is capturing packets and then analyzing the data. The second approach is capturing network flow, also known as NetFlow, as explained in Chapter 4, "NetFlow for Cybersecurity." Both approaches have benefits and disadvantages. Packet capturing can provide more details than NetFlow; however, this approach requires storing packets as well as a method to properly analyze what is captured. Capturing packets can quickly increase storage requirements, making this option financially challenging for some organizations. Other times, the details provided by capturing packets are necessary for establishing baselines as well as security requirements and therefore is the best approach versus what limited data NetFlow can provide. For digital forensics requirements, capturing packets will most likely be the way to go due to the nature of the type of details needed to perform an investigation and to validate findings during the legal process. NetFlow involves looking at network records versus actually storing data packets. This approach dramatically reduces storage needs and can be quicker to analyze. NetFlow can provide a lot of useful data; however, that data will not be as detailed as capturing the actual packets. An analogy of comparing capturing packets	

to NetFlow would be monitoring a person’s phone. Capturing packets would be similar to recording all calls from a person’s phone and spending time listening to each call to determine if there is a performance or security incident. This obviously would be time consuming and require storage for all the calls. Capturing NetFlow would be similar to monitoring the call records to and from the phone being analyzed, meaning less research and smaller storage requirements. Having the phone call (packet capture) would mean having details about the incident, whereas the call record (NetFlow) would show possible issues, such as multiple calls happening at 3 a.m. between the person and another party. In this case, you would have details such as the phone numbers, time of call, and length of call. If these call records are between a married person and somebody who is not that person’s significant other, it could indicate a problem—or it could simply be planning for a surprise party. The point is, NetFlow provides a method to determine areas of concern quickly, whereas packet capturing determines concerns as well as includes details about the event since the actual data is being analyzed versus records of the data when using NetFlow. Also, it is important to note that some vendors offer hybrid solutions that use NetFlow but start capturing packets upon receiving an alarm. One example of a hybrid technology is Cisco’s StealthWatch technology. Once you have your source and data type selected, the final task for establishing a baseline is determining the proper length of time to capture data. This is not an exact science; however, many experts will suggest at least a week to allow for enough data to accommodate trends found within most networks. This requirement can change depending on many factors, such as how the business model of an organization could have different levels of traffic at different times of the year. A simple example of this concept would be how retailers typically see higher amounts of traffic during holiday seasons, meaning a baseline sample during peak and nonpeak business months would most likely be different. Network spikes must be accounted for if they are perceived to be part of the normal traffic, which is important if the results of the baseline are to be considered a true baseline of the environment. Time also impacts results in that any baseline taken today may be different in the future as the network changes, making it important to retest the baseline after a certain period of time. Most network administrators’ goal for understanding throughput is to establish a network baseline so throughput can later be monitored with alarms that trigger at the sign of a throughput valley or peak. Peaks are spikes of throughput that exceed the normal baseline, whereas valleys are periods of time that are below the normal baseline. Peaks can lead to problems, such as causing users to experience long delays when accessing resources, triggering redundant systems to switch to backups, breaking applications that require a certain data source, and so on. A large number of valleys could indicate that a part of the network is underutilized, representing a waste of resources or possible failure of a system that normally utilizes certain resources. Many tools are available for viewing the total throughput on a network. These tools can typically help develop a network baseline as well as account for predicted peaks and valleys. One common metric used by throughput measuring tools is bandwidth, meaning the data rate supported by a network connection or interface. Bandwidth, referred to as bits per second (bps), is impacted by the capacity of the link as well as latency factors, meaning things that slow down traffic performance. Best practice for building a baseline is capturing bandwidth from various parts of the network to accommodate the many factors that impact bandwidth. The most common place to look at throughput is the gateway router, meaning the place that traffic enters and leaves the network. However, throughput issues can occur anywhere along the path of traffic, so only having a sample from the gateway could be useful for understanding total throughput for data leaving and entering the network, but this number would not be effective for troubleshooting any issues found within the network. For example, network congestion could occur between a host and network relay point prior to data hitting the network gateway, making the throughput at the gateway look slower than it actually would be if the administrator only tests for complications

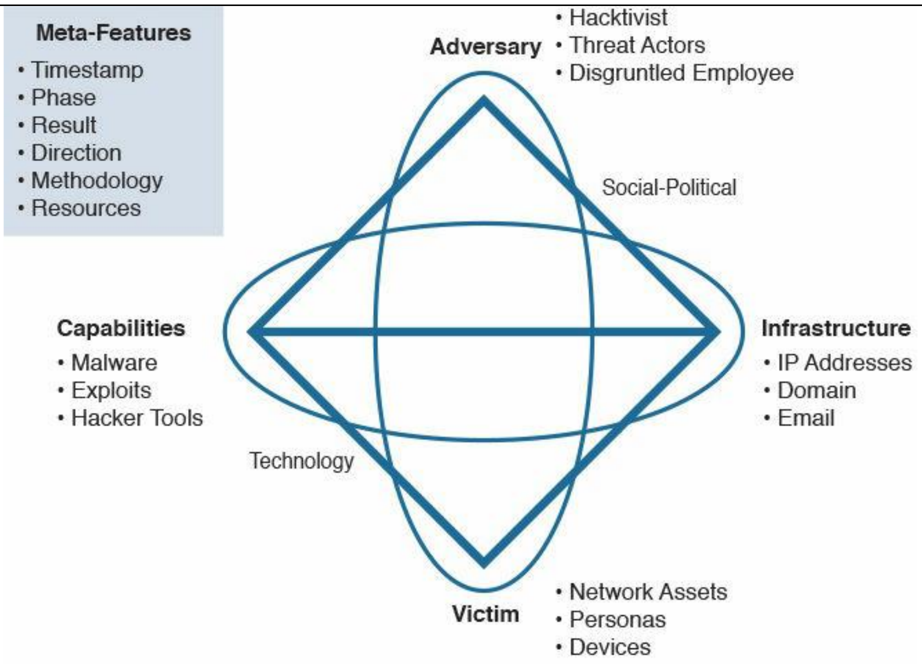
		from the host network and doesn't validate the entire path between the host and gateway. Measuring throughput across the network can lead to the following improvements: • Understanding the impact of applications on the network • Reducing peak traffic by utilizing network optimization tools to accommodate for latency-generating elements such as bandwidth hogs • Troubleshooting and understanding network pain points, meaning areas that cause latency. • Detecting unauthorized traffic • Using security and anomaly detection • Understanding network trends for segmentation and security planning • Validating whether quality of service settings are being utilized properly	
5	**Attempt *any one* of the following:**		**6**
	a)	Explain with an example how mapping threat intelligence with DNS and other artefacts helps in detection of Malicious Activity.	
	A	The Domain Name System (DNS) is a complex network that communicates through multiple servers distributed around the world to match domains with IP addresses. Like a fast-growing city, the rapid expansion of the internet has made DNS vulnerable to attacks, so its traffic needs to be monitored and analyzed to keep users safe. Many security professionals consider SIEM as a storage for all possible logs which are collected from various endpoint, servers, network equipment and software. This allows for the ability to correlate different types of events and effectively identify security threats. However, it doesn't magically happen and may remain an unreachable dream if enough effort is not put into data aggregation and correlation.. Every day thousands of security incidents are left uninvestigated because of lack of resources and automation. So one of the issues is how to effectively identify real threats and prioritize them. Sometimes it's like looking for a needle in a haystack. As mentioned earlier, a SIEM platform contains all kind of events from various software and network equipment, so the indicators can be used to enrich any log messages which contain relevant fields.For example, web-server access logs contain a source (or requestor) IP address and a URL. Both properties can be used in conjunction to detect and mitigate malicious traffic and hunt new threats. The idea is simple – that known HTTP scanners scan for specific software and vulnerabilities. Based on requested URLs we can find out what kind of tools are used and which vulnerabilities malicious actors are trying to exploit. Analyzing similar requests, we can find other hosts that scan our network. Blocking access to such scanners can significantly reduce attack surface.An another example, on our software dashboard which displays bruteforce attempts on my SSH server. The dashboard is powered by threat intelligence engine. Security threat intelligence is extremely useful when correlating events and to gain an insight into what known threats are in your network. DNS intelligence and URL reputation are used in many security solutions such as the Cisco Firepower appliances, Cisco Firepower Threat Defense (FTD), the Cisco Web and Email security appliances, and Cisco Umbrella. For instance, you can correlate security events based on threat intelligence to identify communications to known malicious command and control (CnC) servers based on DNS information.	
	b)	Explain the Extended Diamond Model of Intrusion with an Example.	
	A	The Diamond Model is designed to represent an incident, also called an event, and is made up of four parts. Active intrusions start with an adversary who is targeting a victim. The adversary will use various capabilities along some form of infrastructure to launch an attack against the victim. Capabilities can be various forms of tools, techniques, and procedures, while the infrastructure is	

what connects the adversary and victim. The lines connecting each part of the model depict a mapping of how one point reached another. For example, the analyst could see how a capability such as a phishing attack is being used over an infrastructure such as email and then relate the capabilities back to the adversary. Moving between each part of an attack is called analytic pivoting and is key for modeling the event.

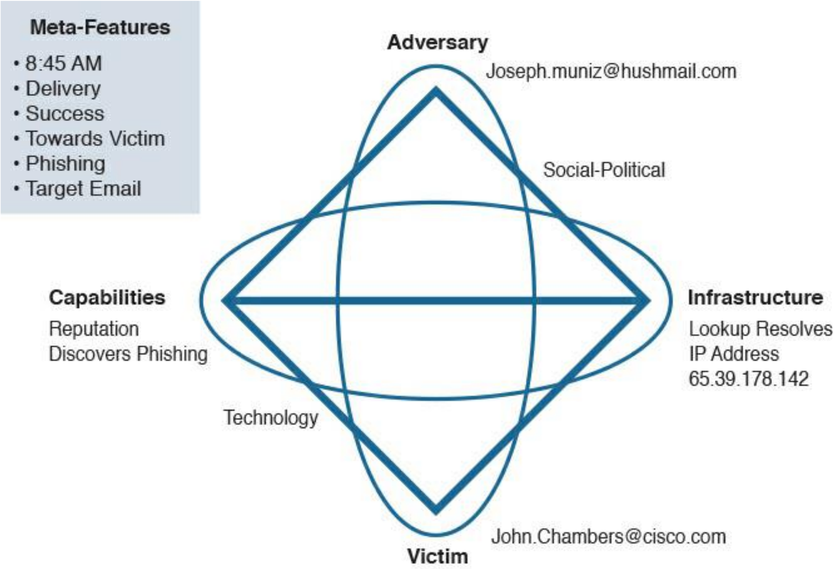
The Diamond Model also includes additional meta-features of an event, such as a timestamp, kill chain phase, result of the attack, direction of the attack, attack method, and resources used. An example of a meta-feature list might show a timestamp of 1:05 p.m. The kill chain phase could be exploitation, the result could be successful, the direction could be adversary to victim, the attack method could be spear-phishing, and the resources could be a specific vulnerability on the victim's host system. Meta-features provide useful context but are not core to the model, so they can be disregarded and augmented as necessary.



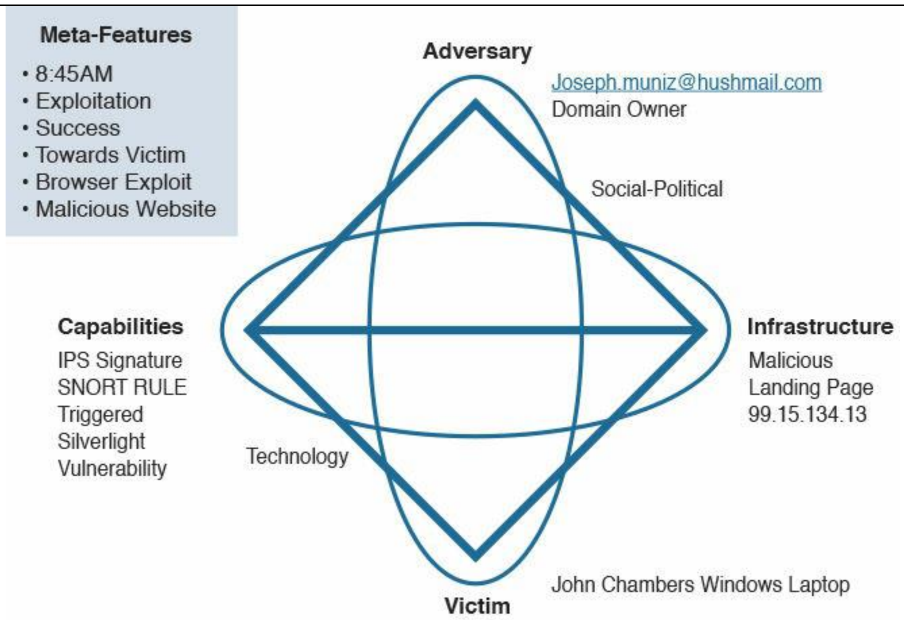
The Diamond Model can be expanded further by adding two additional metafeatures that establish connections between relations. The technology metafeature connects capabilities and infrastructure by describing the technology used between these two parts of the model. An example of a technology metafeature could be the domain name system (DNS) if it is used by malware to determine its command-and-control point. The social-political meta-feature represents the relationship between the adversary and victim. This is critical to determine the intent behind the attack so the analyst can understand the reason the victim was selected and the value the adversary sees in the victim, as well as sometimes identify a shared threat space, meaning a situation where multiple victims link back to the same adversaries. A shared threat space equates to threat intelligence—that is, understanding threat actors in a specific space to potentially forecast and react to future malicious activity. An example might be threat actors identified for launching an attack campaign against hospitals.



Each event is considered a diamond using this model. An incident management practice should use the Diamond Model as the basis for grouping and organizing incidents. The goal would be to review multiple diamonds and identify a common adversary. For example, let's consider an attack where the adversary is delivering ransomware to a victim. The first part of the attack could involve the adversary using a malicious email to trick the victim into accessing a website. The goal is to have the website scan the victim for vulnerabilities and deliver ransomware by exploiting one of those weaknesses.



Stage 2 of the attack follows the phishing email that redirected the victim's system to the malicious website. Now that the victim's system has accessed the website, the malicious website will push down the ransomware by exploiting a vulnerability. The adversary is still the same attacker; however, the capabilities and infrastructure involved with the second part of the security incident have changed, which is common when identifying all stages of an attack according to the kill chain concept.



Instances of the same event occurring over the course of a few weeks could be linked together through multiple diamonds and then linked back to the same adversary. Linking the spear-phishing attack to the delivery of ransomware can give an analyst a method to diagram the attack and all associated adversaries. The incident response team should create an activity group based on the various connected diamonds and attempt to define what combinations of elements are criteria for grouping diamonds together. As new diamonds appear, activity groups can grow as diamonds are grouped together based on newly available data. The relationships between diamonds are known as activity threads, which can spread across the same attack as well as connect other attacks, depending on intelligence gathered that meets activity group requirements.

c) What are different network scanning techniques. Discuss in brief.

A There are different types of port- and network-scanning techniques. The following are the most common:

Basic port scan: Involves scanning a predetermined TCP/UDP port by sending a specifically configured packet that contains the port number of the port that was selected. This is typically used to determine what ports are "open" or available in a given system.

TCP scan: A TCP-based scan of a series of ports on a machine to determine port availability. If a port on the machine is listening, then the TCP "connect" is successful in reaching that specific port. Earlier you learned that nmap is an open source scanner; nmap refers to TCP scans as "connect scans," which is named after the UNIX connect() system call. If the scanner finds that a port is open, the victim operating system completes the TCP three-way handshake. In some cases, the port scanner will close the connection to avoid a denial-of-service condition. TCP SYN scan is one of the most common types of TCP scanning, and it is also referred to as "half-open scanning" because it never actually opens a full TCP connection. The scanner sends a SYN packet, and if the target responds with a SYN-ACK packet, the scanner typically responds with an RST packet. Another TCP scan type is TCP ACK. This type of scan does not exactly determine whether the TCP port is open or closed; instead, it checks whether the port is filtered or unfiltered. TCP ACK scans are typically used when trying to see if a firewall is deployed and its rule sets. There are also TCP FIN packets that in some cases can bypass legacy firewalls because closed ports may cause a system to reply to a FIN packet with a corresponding RST packet due to the nature of TCP.

		UDP scan: Because UDP is a connectionless protocol and does not have a threeway handshake like TCP, the UDP scans have to rely on ICMP “port unreachable” messages to determine if the port is open. When the scanner sends a UDP packet and the port is not open on the victim, the victim’s system will respond with an ICMP “port unreachable” message. This type of scanning will be affected by firewalls and ICMP rate limiting. **Strobe scan**: Typically used by an attacker to find the ports that he or she already knows how to exploit. Strobe scans execute on a more confined level. **Stealth scan:** Designed to go undetected by network auditing tools.	