VIGNAN'S LARA
INSTITUTE OF TECHNOLOGY & SCIENCE
Approved by AICTE New Delhi & Affiliated to JNTUK Kakinada
Accredited by NAAC 'A+' and NBA | ISO 9001 : 2015
Vadlamudi - 522 213, Guntur District

SECURE CODING TECHNIQUES

QUESTION BANK

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

UNIT-I

Network and Information security Fundamentals

1) What are the different types of network models?

A) Computer Network Models

A communication subsystem is a complex piece of Hardware and software. Early attempts for implementing the software for such subsystems were based on a single, complex, unstructured program with many interacting components. The resultant software was very difficult to test and modify. To overcome such problem, the ISO has developed a layered approach. In a layered approach, networking concept is divided into several layers, and each layer is assigned a particular task. Therefore, we can say that networking tasks depend upon the layers.

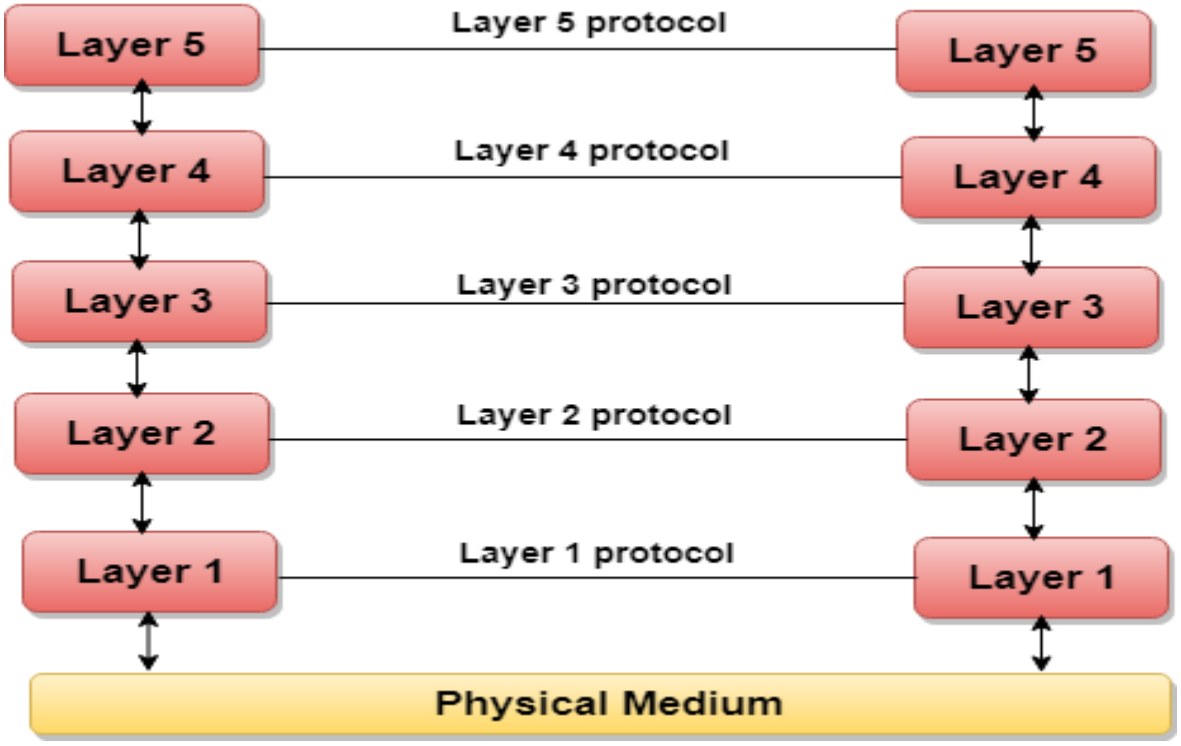
Layered Architecture

- The main aim of the layered architecture is to divide the design into small pieces.
- Each lower layer adds its services to the higher layer to provide a full set of services to manage communications and run the applications.
- It provides modularity and clear interfaces, i.e., provides interaction between subsystems.
- It ensures the independence between layers by providing the services from lower to higher layer without defining how the services are implemented. Therefore, any modification in a layer will not affect the other layers.
- The number of layers, functions, contents of each layer will vary from network to network. However, the purpose of each layer is to provide the service from lower to a higher layer and hiding the details from the layers of how the services are implemented.
- The basic elements of layered architecture are services, protocols, and interfaces.
    - **Service:** It is a set of actions that a layer provides to the higher layer.

- **Protocol:** It defines a set of rules that a layer uses to exchange the information with peer entity. These rules mainly concern about both the contents and order of the messages used.
- **Interface:** It is a way through which the message is transferred from one layer to another layer.

- In a layer n architecture, layer n on one machine will have a communication with the layer n on another machine and the rules used in a conversation are known as a layer-n protocol.

Let's take an example of the five-layered architecture.



- In case of layered architecture, no data is transferred from layer n of one machine to layer n of another machine. Instead, each layer passes the data to the layer immediately just below it, until the lowest layer is reached.
- Below layer 1 is the physical medium through which the actual communication takes place.
- In a layered architecture, unmanageable tasks are divided into several small and manageable tasks.
- The data is passed from the upper layer to lower layer through an interface. A Layered architecture provides a clean-cut interface so that minimum information is shared among different layers. It also ensures that the implementation of one layer can be easily replaced by another implementation.
- A set of layers and protocols is known as network architecture.

2) What are the three components of network security?

a) Communication Networks can be of following 5 types:

1. Local Area Network (LAN)

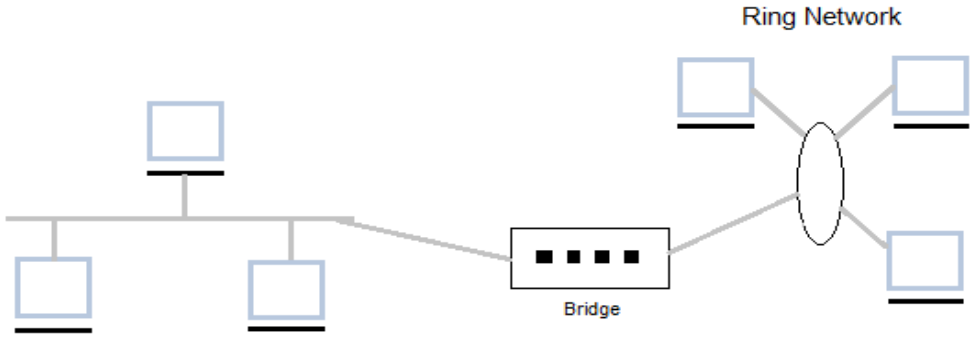
2. Metropolitan Area Network (MAN)
3. Wide Area Network (WAN)

Local Area Network (LAN)

It is also called LAN and designed for small physical areas such as an office, group of buildings or a factory. LANs are used widely as it is easy to design and to troubleshoot. Personal computers and workstations are connected to each other through LANs. We can use different types of topologies through LAN, these are Star, Ring, Bus, Tree etc.

LAN can be a simple network like connecting two computers, to share files and network among each other while it can also be as complex as interconnecting an entire building.

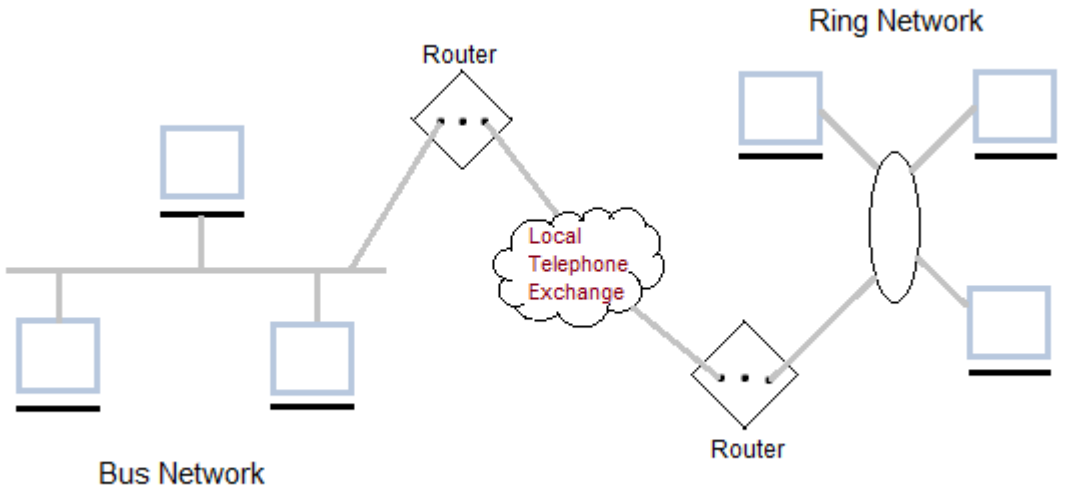
LAN networks are also widely used to share resources like printers, shared hard-drive etc.



(Different Topologies interconnected in a Local Area Network)

Metropolitan Area Network (MAN)

It was developed in 1980s.It is basically a bigger version of LAN. It is also called MAN and uses the similar technology as LAN. It is designed to extend over the entire city. It can be means to connecting a number of LANs into a larger network or it can be a single cable. It is mainly hold and operated by single private company or a public company.



Characteristics of MAN

- It generally covers towns and cities (50 km)
- Communication medium used for MAN are optical fibers, cables etc.
- Data rates adequate for distributed computing applications.

Wide Area Network (WAN)

It is also called WAN. WAN can be private or it can be public leased network. It is used for the network that covers large distance such as cover states of a country. It is not easy to design and maintain. Communication medium used by WAN are PSTN or Satellite links. WAN operates Characteristics of WAN on low data rates.

- It generally covers large distances(states, countries, continents).
- Communication medium used are satellite, public telephone networks which are connected by routers.

3) What are the three security objectives in cyber security? What are the core security objectives?

Cyber Security Objectives and Services

Understanding Cyber Security

Cyber security is a complex field that requires a deep understanding of the various threats that exist and the measures that can be taken to protect against them. It is important to understand the different types of threats that exist, such as malware, phishing, and ransom ware, and the different types of security measures that can be taken to protect against them. This includes the use of firewalls, antivirus software, and other security measures. It is also important to understand the different types of data that need to be protected, such as customer data, financial data, and intellectual property.

Implementing Cyber Security Measures

Once an organization has a good understanding of the threats that exist and the measures that can be taken to protect against them, it is important to implement the necessary security measures. This includes the use of firewalls, antivirus software, and other security measures. It is also important to ensure that all employees are trained on the proper use of these security measures and that they are aware of the risks associated with not following security protocols. Additionally, organizations should regularly review their security measures to ensure that they are up to date and effective.

Monitoring Cyber Security

In addition to implementing security measures, organizations should also monitor their networks and systems for any suspicious activity. This includes monitoring for any unauthorized access to the network or any suspicious activity on the network. Organizations should also monitor for any changes to the system or any suspicious activity on the system. Additionally, organizations should regularly review their security measures to ensure that they are up to date and effective.

Responding to Cyber Security Incidents

When a cyber security incident occurs, it is important for organizations to respond quickly and effectively. This includes identifying the source of the attack, assessing the damage, and taking steps to mitigate the damage. It is also important to notify the appropriate authorities and take steps to prevent similar incidents from occurring in the future. Additionally, organizations should review their security measures to ensure that they are up to date and effective.

4) What are network models in cyber security?

A) Computer Network Models

A communication subsystem is a complex piece of Hardware and software. Early attempts for implementing the software for such subsystems were based on a single, complex, unstructured program with many interacting components. The resultant software was very difficult to test and modify. To overcome such problem, the ISO has developed a layered approach. In a layered approach, networking concept is divided into several layers, and each layer is assigned a particular task. Therefore, we can say that networking tasks depend upon the layers.

Layered Architecture

- The main aim of the layered architecture is to divide the design into small pieces.
- Each lower layer adds its services to the higher layer to provide a full set of services to manage communications and run the applications.
- It provides modularity and clear interfaces, i.e., provides interaction between subsystems.
- It ensures the independence between layers by providing the services from lower to higher layer without defining how the services are implemented. Therefore, any modification in a layer will not affect the other layers.
- The number of layers, functions, contents of each layer will vary from network to network. However, the purpose of each layer is to provide the service from lower to a higher layer and hiding the details from the layers of how the services are implemented.
- The basic elements of layered architecture are services, protocols, and interfaces.
    - Service: It is a set of actions that a layer provides to the higher layer.
    - Protocol: It defines a set of rules that a layer uses to exchange the information with peer entity. These rules mainly concern about both the contents and order of the messages used.
    - Interface: It is a way through which the message is transferred from one layer to another layer.
- In a layer n architecture, layer n on one machine will have a communication with the layer n on another machine and the rules used in a conversation are known as a layer-n protocol.

5) Discuss the Common Cyber security Myths with examples?

A) **common cyber security myths**

1. Security software slows down or interrupts workflow

this one simply isn't true, and we believe the myth has originated from poor implementation of security tools, rather than the limitations of the tools themselves. if security tools have been implemented properly then you should be provided with security without affecting your users' productivity.

2. i have a strong password, i am safe

While having a strong password is a necessity, unfortunately it isn't enough on its own. a good way to add another level of security is to use multi-factor authentication (mfa), requiring users to authenticate themselves via a second method such as their phone or an app like google authenticator. with mfa in place, even if criminals do manage to get hold of usernames and passwords, they still won't be able log in without the 'second factor'.

3. security costs too much

companies who think like this are often not considering the downside costs. data breaches will end up being much costlier to your business than making sure you have dedicated security solutions in place before they can happen. capita estimates the average cost of a data breach to be $3.86 million, considering the cost of detecting and escalating a breach, notifying those affected and the regulatory authorities, lost business and reputational damage, and paying fines, legal fees and other costs associated with making things right.

4. i will know straight away if my business is attacked

this rarely the case these days. there used to be some easy signs (pop up ads or slow loading browsers) but scammers have become stealthier. hacking is a silent crime and it is in criminals' best interest to remain unnoticed for as long as possible. the longer they have access to your systems, the more data they can steal.

5. Cyber security is solely the it department's responsibility

unfortunately, neglectful employees are the number one cause of cyber security breaches, so you can't rely solely on the it department to keep your organization secure online – everyone has a role to play. all your staff should be using corporate laptops/tablets/phones with at least 2 factor authentication, as well as ensuring that their installed security software is up to date.

6) What is the general conclusion of cyber security?

A) **1. Antivirus and Cyber-Security Software is Good Enough**

There's a lot that can go wrong here. Although most people feel at ease after installing security software, they're not nearly air-tight in reality. The servers of such security software providers are vulnerable to hacking attacks, rendering the clients' defenses useless.

The kind of cyber-security software you choose is also important. It's easy to select an antivirus at random and live to regret it later. Always go with reliable providers with stronger safeguards. Some great ones might charge a buck or two, but curtailing costs here might cost you big-time in the future.

2. Complex Passwords Cannot Be Cracked

Passwords are becoming really easy to breach for hackers. Special programs are capable of cracking the longest and most confusing passwords by trying billions of different combinations in the space of seconds. Password trends can also be further replicated to breach your security in multiple online

avenues, e.g., having a password for a social media site and using the same one for your email account.

Temporary passwords, OTPs, and two-factor authentication are a way to reduce the risk.

3. My Data Isn't Worth Anything

That is not true. If it were, social media would never be free to begin with. If a service such as that is free, it monetizes your data instead, selling it to advertisers as an entire 'customer' profile.

Data can be materialized for crime, such as theft, impersonation, and physical harm. If it's valuable for some, it's valuable for many.

4. Scams and Phishing Are Glaringly Obvious

Phishing schemes and scams are getting more and more intelligent and convincing. Some pretend to withhold your sensitive information via webcam and threaten to release it. Others masquerade as services that you are currently subscribed to and give 'reminders' about privacy settings updates.

The data they used to reach you, such as the email address and password, has probably been breached. Some hackers even manage to breach the social media accounts of people you know and use your trust in them against you by sending links to malicious content.

5. Mainstream Websites Are Safe to Visit

All of the big websites employ cookies to track your internet trajectory. Despite the onsite safety, these companies that own the sites possess your data. If any of these companies are hacked, your data is breached too. Consequently, your data isn't as safe with big websites either.

7) Evaluate the recent major cyber attacks? How do cyber attacks happen?

A) Recent Major Cyber attacks are

Bank Accounts Hacked in Nepal

- **Date:** February 3, 2023
- **Attack type:** credential theft
- **Target:** Individuals using net banking
- **Impact:** Several million rupees stolen

Eight malicious actors were arrested in Kathmandu, Nepal, were arrested by the police for hacking into bank accounts. The attackers shared the Android package kit (APK) for a fake app called Nepali Keti over WhatsApp. Then they hacked into the bank accounts of the people who downloaded the app and stole money.

Dish Network faced a data breach

- **Date:** February 23, 2023
- **Attack type:** Data Breach
- **Target:** Dish Network
- **Impact:** Some data was extracted and Dish's share price fell by 6.5%

Dish Network, one of the USA's biggest television providers, disclosed that the network outage reported earlier was connected to a cyber attack. The root causes of the intrusion are yet to be found. The attack resulted in data theft and internal communication breakdown.

US Marshals Service faces ransomware attack

- **Date:** February 17, 2023
- **Attack type:** Ransomware
- **Target:** USMS
- **Impact:** Sensitive law enforcement data exposed

The U.S. Marshals Service is responsible for sensitive tasks like the security of federal judges, fugitive apprehension, etc. The stand-alone USMS system was compromised by attackers exposing data related to USMS investigations.

8) What is cyber security in basic terms?

A) Credential stuffing is a cyber-attack where hackers use automated tools to enter thousands of user IDs and passwords stolen during earlier attacks into the input fields meant for customers. Due to the habit of people using the same credentials for multiple accounts, credential stuffing actually works.

In the case of PayPal users, hackers had access to the full names, dates of birth, social security numbers, postal addresses, and individual tax identification numbers of 34,942 users for 2 days.

As cyber-attacks volume and complexity increase, cyber security's importance also increases. Cyber security is critical because it helps to protect organizations and individuals from cyber attacks. Cyber security can help to prevent data breaches, identity theft, and other types of cybercrime. Organizations must have strong cyber security measures to protect their data and customers.

1. Technology Innovation

The importance of cyber security regarding technology innovation is that it helps protect ideas and intellectual property from theft or being copied without permission. This is important because it allows companies to maintain a competitive advantage and keep their products and services safe

from competitors. Additionally, it helps to ensure that new products and services are not easily replicated or stolen before they can be released to the market.

2. Cloud Transformation

The cloud has transformed how we think about IT, but it has also introduced new security risks. As organizations move more critical data and applications to the cloud, they must know the latest cyber security threats and how to protect themselves.

One of the most significant advantages of the cloud is that it allows organizations to be more agile and responsive to change. However, this agility can also introduce new security risks. For example, a cloud provider may not have the same security controls as a traditional on-premises data center. Cloud data is often spread across multiple physical locations, making protecting it more challenging.

3. Maintaining Customer and Employee Trust

Customers and employees trust that their information will be protected from cyber threats. To maintain this trust, businesses must invest in cyber security measures to protect customer and employee data. This may include installing firewalls, encrypting data, and creating secure passwords. By taking these steps, businesses can show their commitment to protecting customer and employee information, which can help to build and maintain trust.

4. Securing Financial Position of the Organization

The importance of cyber security to ensure an organization's financial position cannot be understated. In today's interconnected world, where sensitive data is often stored digitally, a breach in security can have disastrous consequences. Not only can it lead to the loss of crucial data, but it can also damage an organization's reputation and bottom line. A cyber attack can result in the loss of customer confidence, increased costs, and a drop in stock value.

9) What is aim of cyber security? What are the three goals of cyber security?

A) Cyber Security Objectives and Services

Understanding Cyber Security

Cyber security is a complex field that requires a deep understanding of the various threats that exist and the measures that can be taken to protect against them. It is important to understand the different types of threats that exist, such as malware, phishing, and ransom ware, and the different types of security measures that can be taken to protect against them. This includes the use of firewalls, antivirus software, and other security measures. It is also important to understand the different types of data that need to be protected, such as customer data, financial data, and intellectual property.

Implementing Cyber Security Measures

Once an organization has a good understanding of the threats that exist and the measures that can be taken to protect against them, it is important to implement the necessary security measures. This includes the use of firewalls, antivirus software, and other security measures. It is also important to ensure that all employees are trained on the proper use of these security measures and that they are aware of the risks associated with not following security protocols. Additionally, organizations should regularly review their security measures to ensure that they are up to date and effective.

Monitoring Cyber Security

In addition to implementing security measures, organizations should also monitor their networks and systems for any suspicious activity. This includes monitoring for any unauthorized access to the network or any suspicious activity on the network. Organizations should also monitor for any changes to the system or any suspicious activity on the system. Additionally, organizations should regularly review their security measures to ensure that they are up to date and effective.

10) What are the three categories of computer attacks?

A) Types of Cyber Attacks

There are many varieties of cyber attacks that happen in the world today. If we know the various types of cyber attacks, it becomes easier for us to protect our networks and systems against them. Here, we will closely examine the top ten cyber-attacks that can affect an individual, or a large business, depending on the scale.

Let’s start with the different types of cyber attacks on our list:

1. Malware Attack

This is one of the most common types of cyber attacks. “Malware” refers to malicious software viruses including worms, spyware, ransom ware, adware, and Trojans.

The trojan virus disguises itself as legitimate software. Ransom ware blocks access to the network's key components, whereas Spyware is software that steals all your confidential data without your knowledge. Adware is software that displays advertising content such as banners on a user's screen.

Malware breaches a network through vulnerability. When the user clicks a dangerous link, it downloads an email attachment or when an infected pen drive is used.

Let’s now look at how we can prevent a malware attack:

- Use antivirus software. It can protect your computer against malware. Avast Antivirus, Norton Antivirus, and McAfee Antivirus are a few of the popular antivirus software.
- Use firewalls. Firewalls filter the traffic that may enter your device. Windows and Mac OS X have their default built-in firewalls, named Windows Firewall and Mac Firewall.
- Stay alert and avoid clicking on suspicious links.
- Update your OS and browsers, regularly.

2. Phishing Attack

Phishing attacks are one of the most prominent widespread types of cyber attacks. It is a type of social engineering attack wherein an attacker impersonates to be a trusted contact and sends the victim fake mails.

Unaware of this, the victim opens the mail and clicks on the malicious link or opens the mail's attachment. By doing so, attackers gain access to confidential information and account credentials. They can also install malware through a phishing attack.

Phishing attacks can be prevented by following the below-mentioned steps:

- Scrutinize the emails you receive. Most phishing emails have significant errors like spelling mistakes and format changes from that of legitimate sources.
- Make use of an anti-phishing toolbar.
- Update your passwords regularly.

3. Password Attack

It is a form of attack wherein a hacker cracks your password with various programs and password cracking tools like Air crack, Cain, Abel, John the Ripper, Hash cat, etc. There are different types of password attacks like brute force attacks, dictionary attacks, and keylogger attacks.

Listed below are a few ways to prevent password attacks:

- Use strong alphanumeric passwords with special characters.
- Abstain from using the same password for multiple websites or accounts.
- Update your passwords; this will limit your exposure to a password attack.

Do not have any password hints in the op

UNIT-II

Introduction to Cyber security

1) What are the main mitigation measures for injection attacks?

A) SQL Injection

The goal of an injection attack is to inject SQL, NoSQL, OS, and LDAP data into the application. It can be done through the application's input interface as SQL queries. If SQL injection is successful, the database's sensitive data may be exposed.

SQL injection can be used to edit database data using Insert, Update, and Delete statements, as well as shut down the DBMS (Database Management System) with merely a SQL injection.

Because of the lack of input validation and data sanitization, which might directly expose input into the query, injection happens when data is entered into a program from an un trusted source. This injection vulnerability may be found on practically any website, demonstrating how serious it is. Anything that accepts parameters as input can be vulnerable to injection.

Broken Authentication

Broken authentication is one of the OWASP top 10 significant vulnerabilities, which attackers can employ to impersonate a valid user online.

Session management and credential management are the two locations where this vulnerability is always present. These two are classified as broken authentication since they can both be used to steal login credentials or hijack session IDs. Attackers use a variety of techniques to exploit these flaws, ranging from credential stuffing to other highly targeted methods of gaining unauthorized access to someone's credentials.

Exposed Sensitive Data

This is one of the OWASP Top 10 vulnerabilities for data compromise that requires protection. This is often referred to as information disclosure or leakage. This commonly happens when a program or website unintentionally releases sensitive information to people who do not have permission to see or access it.

These are some of the details that could be released to the public, according to OWASP −

- Information about money
- Login information
- Data that is commercial or business-related
- A medical history
- Technical information about the app or website

Even if you aren't utilizing DEBUG=True, you must exercise caution while managing the configuration parameter.

XXE Injection

XML external entity injection (also known as XXE) is a security flaw that allows a malicious person to get access to an application that processes XML data or parses XML input, according to the OWASP.

Because XML input containing a reference to an external entity is handled by an XML parse that has been configured incorrectly, this attack is always effective. An attacker can examine files on the application server and interact with any other back end or external system that the application can access if this vulnerability is successfully exploited.

Through Server Side Request Forgery (SSRF) attacks, this XXE attack can be used to compromise other back-end or underlying systems.

The vulnerability is not in the data you give to the server in XML format; rather, it is in the way the XML is parsed.

When XML parsers that support DTD retrieval do not have sufficient input validation of the XML data in place, they may be vulnerable to XXE injection, which allows an attacker to inject commands or content into an XML document.

2) What is cross-site scripting XSS?

A) The technique of inserting malicious code on a legitimate website in order to capture user information for nefarious reasons is known as cross-site scripting (XSS).

- By inserting malicious code in a genuine web page or web application, the attacker attempts to execute harmful scripts on the victim's web browser.
- The real attack occurs when the victim visits a website or uses a web application containing malicious code. The malicious script is delivered to the user's browser via the web page or application.
- Forums, message boards, and online pages with commenting capabilities are typical targets for Cross-site Scripting attacks.

When a web page or a web application produces output that incorporates un sanitized user input, it is known as XSS. This user input must be parsed by the victim's browser.

- XSS attacks can be carried out in VBScript, ActiveX, Flash, and even CSS. They are, nevertheless, most ubiquitous in JavaScript because JavaScript is essential to most browser experiences.
- The security of that vulnerable website or online application, as well as its users, has been compromised if an attacker can exploit XSS vulnerability on a web page to run arbitrary JavaScript in a user's browser.
- Cross-site scripting can be used to deface a website. The attacker can use injected scripts to alter the website's content or even redirect the browser to another website, such as one that includes malicious code.
- **Types of Cross-site Scripting (XSS) Attacks**
- XSS attacks come in a variety of forms. They are classified into the following categories −
- **Persistent XSS**
- This sort of vulnerability, also known as stored XSS, happens when un trusted or unverified user input is stored on a target server. Message boards, comment sections, and visitor logs are all common targets for persistent XSS—any feature where other users, both authenticated and unauthenticated, would see the attacker's malicious text.

- **Reflective XSS**
- Reflected or non-persistent cross-site scripting, on the other hand, entails the immediate return of user input. An attacker must mislead the user into transferring data to the target site to exploit a reflected XSS, which is generally done by deceiving the user into clicking a maliciously designed link.

3) What is mitigation of broken authentication? What are the effects of broken authentication?

A) Broken Access Control

Broken access control refers to a situation in which an attacker is able to access a resource or perform an action that should be restricted. This can happen for a number of reasons. For example, a developer may have inadvertently made a security mistake, or vulnerability may have been introduced due to a third-party library or service.

There are many different ways in which access control can be broken. For example, an attacker may be able to bypass authentication, gain access to privileged data or resources, or modify data that should be read-only. Broken access control can occur at any level of an application, from the user interface to the back-end data storage.

What are the Consequences of Broken Access Control?

The consequences of broken access control can be severe. In some cases, an attacker may be able to gain full control of a system, steal sensitive data, or delete important information. For example, an attacker might be able to bypass a login screen and gain access to a user's account. Once inside, the attacker could steal sensitive information, modify account settings, or even initiate fraudulent transactions.

In other cases, the damage caused by broken access control may be more subtle. For example, an attacker might be able to gain access to sensitive information that they are not authorized to see, leading to privacy violations or regulatory noncompliance. Broken access control can also lead to reputational damage, loss of customer trust, and other business risks.

Preventing broken access control requires a multi-layered approach that involves careful design, development, and testing. In the following sections, we'll explore some of the key techniques and best practices that can help prevent broken access control.

Access Control Design

One of the most important steps in preventing broken access control is to design access control measures that are robust and effective. Access control design should consider the different roles and levels of access that users or entities will need, and ensure that the appropriate restrictions are in place.

For example, it's important to limit access to administrative functions to only those users who need it, and to ensure that access is granted based on the principle of least privilege. This means that users are given only the minimum level of access that they need to perform their job, and no more. This can help prevent situations where a user with too much access accidentally or intentionally causes harm to the system.

4) What are the different types of XXE?

A) XML External Entity attack

XXE or XML External Entity attack is a web application vulnerability that affects a website which parses unsafe XML that is driven by the user. XXE attack when performed successfully can disclose local files in the file system of the website. XXE is targeted to access these sensitive local files of the website that is vulnerable to unsafe parsing.

Types of XXE Attacks

1. **File Retrieval XXE**: As the name implies, arbitrary files on the application server of a victim company can be exposed to the attacker, if there is an XXE vulnerable endpoint in the target system. This can be carried out by passing an external XML entity in the user-controlled file.
2. **Blind XXE:** It is possible that a target system doesn't return data from the entities placed by the attacker still being insecure and vulnerable to XXE. This is done by trying out malformed user inputs. These include the input of length more than what the system expects the wrong data type, special entities, etc. The intention is to make the system fail and check if throws out some sensitive information in the error response.

XXE to SSRF: Even if the system doesn't return the response with local file content to the attacker, the system can be still exploited in presence of an XXE attack. The entity can be pointed to a local IP of the target company which can be accessed only by its websites/network. Placing an intranet IP in XXE payload will make the target application call its local endpoint which the attacker won't have access to otherwise. This type of attack is called SSRF or Server Side Request Forgery.

5) Describe the missing function level access control mitigation?

A) The missing function level access control vulnerability allows users to perform functions that should be restricted, or lets them access resources that should be protected. Normally, functions and resources are directly protected in the code or by configuration settings, but it's not always easy to do correctly.

Risk mitigation involves a systematic process in which an organization identifies, analyses, and proposes measures to effectively handle various risks that might pose threat to an organization's functions or operations. Every organization must carefully curate its risk mitigation plan.

PMBOK offers the following focus areas or procedures for an effective risk management plan

- Plan a risk management process
- Identify potential risks
- Conduct a qualitative risk analysis
- Conduct a quantitative risk analysis
- Plan responses for risks
- Implement risk responses
- Monitor the potential risks

Various Risk Mitigation Strategies:

Depending on the organization and the type of project, the risk mitigation strategies might differ. Sometimes one strategy might be preferably used when compared to others. While in some cases, a combination of strategies can be used depending on the magnitude of the problem at hand. Here are the different types of approaches that can be followed to mitigate risks in a project.

6) How does the cross site request forgery CSRF attack work?

A) CSRF in action

A web application is vulnerable to CSRF if it relies on session cookies to identify users, and doesn't have any other mechanism for validating requests. Additionally, the request we want to exploit needs to have predictable request parameters so that we can create our own request, like the one in this example.

Luckily for us, we've been tipped off about a vulnerable banking application called "Saturn Bank". This app simply uses session cookies to verify requests. Additionally, the session cookies don't have the Same Site attribute. We'll deep dive into what this attribute is later, in short they control how cookies are submitted in cross site requests.

Components with known vulnerabilities :

- These components can be defined as the third-party apps or software or platforms that are outdated and contain bugs that are public to all, that is- sites like https://www.exploit-db.com contain the full detail as to how to exploit the bugs to put the security of the whole website under severe threat.
- This vulnerability arises with the fact that a website finds it difficult to code everything while making its website functional like -transaction, location, chats, etc.
- So, in order to ease the process of building the website, many websites use third-party apps which do the tasks. But the main problem is that those apps can be harmful to their system if they are not updated regularly. Components with known vulnerabilities are considered to be one of the top 10 web application vulnerabilities listed by OWASP.
- Many websites security gets compromised when they use components having known vulnerabilities.

How to spot:
There are automated tools available online ex- drop scanner which display all the outdated components present in the software. These automated tools save time for security experts by directly pointing out flaws of the website.

Business impact:
When a website uses such outdated software, the hacker can misuse the flaws and get inside the database of the website and can cause serious damage to the website by stealing critical information.

7) How can the risk of insecure deserialization attacks be reduced by avoiding?

A) Botnet Attacks

Attackers often leverage several devices connected to the internet to inject malware into a system and coordinate a cyber attack. Such malware is *automated bots* that manipulate the application in different ways – from simple spamming operations to performing more complex attacks intended to manipulate the application.

These are also commonly supported by *botnets* that orchestrate various attacks, including Brute Force, Phishing, and Distributed Denial of Service (DDoS) attacks. Botnet attacks rely on a chain of actions running through multiple stages. In the absence of proper logging of event data, these attacks are almost impossible to detect or analyze.

An efficient monitoring system with tools like Syslog is often considered the primary first line of defense to reduce the likelihood and severity of Botnet attacks.

DNS Attacks

A Domain Name Service (DNS) offers a standard mechanism to point machine hostnames to their IP addresses. Since DNS directs network traffic towards the correct web servers and target machines, these are common vulnerable points that are often exploited by attack vectors to target the availability or stability of the DNS server as part of the overall attack strategy.

Some possible DNS attacks include:

- Cache poisoning
- Distributed Reflection DoS Attacks
- NXDOMAIN attacks
- DNS Tunneling

If DNS-based events are not logged and appropriately monitored, administrators won't know the types of machines attackers (in the disguise of users) query and interact with. Additionally, threat actors can perpetuate malicious actions such as malware installation, credential theft, command & control communication, network foot printing, and data theft in the absence of adequate query logging and analysis.

Insider Threats

Organizations that typically invest a fortune in securing systems from external attacks often miscalculate internal threats. Such internal threat actors continue to be a critical concern for organizations since their suspicious activities often go unchecked. In such cases, malicious or compromised insiders pose a severe threat to systems since they have access to various control and security measures. Though a situation like this sounds astonishing, the mitigation is relatively straightforward and straightforward and relies on an efficient logging mechanism.

Insufficient monitoring and log management in such instances result in untraceable user behavior patterns, thereby allowing imposters or malicious insiders to compromise the system at a much deeper level.

Some commonly known insider threats arising from insufficient logging & monitoring include:

- Malware traffic
- Ransom ware attacks
- Advanced Persistent Threat

8) Which risk is associated with security logging and monitoring failures?

A) Threats Associated with Insufficient Logging & Monitoring

Botnet Attacks

Attackers often leverage several devices connected to the internet to inject malware into a system and coordinate a cyber attack. Such malware is *automated bots* that manipulate the application in different ways – from simple spamming operations to performing more complex attacks intended to manipulate the application.

These are also commonly supported by *botnets* that orchestrate various attacks, including Brute Force, Phishing, and Distributed Denial of Service (DDoS) attacks. Botnet attacks rely on a chain of actions running through multiple stages. In the absence of proper logging of event data, these attacks are almost impossible to detect or analyze.

An efficient monitoring system with tools like Syslog is often considered the primary first line of defense to reduce the likelihood and severity of Botnet attacks.

DNS Attacks

A Domain Name Service (DNS) offers a standard mechanism to point machine hostnames to their IP addresses. Since DNS directs network traffic towards the correct web servers and target machines, these are common vulnerable points that are often exploited by attack vectors to target the availability or stability of the DNS server as part of the overall attack strategy.

Some possible DNS attacks include:

- Cache poisoning
- Distributed Reflection DoS Attacks
- NXDOMAIN attacks
- DNS Tunneling
- Random Sub domain Attacks
- Domain lock-up attack

Insider Threats

- Organizations that typically invest a fortune in securing systems from external attacks often miscalculate internal threats. Such internal threat actors continue to be a critical concern for

organizations since their suspicious activities often go unchecked. In such cases, malicious or compromised insiders pose a severe threat to systems since they have access to various control and security measures. Though a situation like this sounds astonishing, the mitigation is relatively straightforward and straightforward and relies on an efficient logging mechanism.

- Insufficient monitoring and log management in such instances result in untraceable user behavior patterns, thereby allowing imposters or malicious insiders to compromise the system at a much deeper level.

9) What are components with known vulnerabilities?

A) Components with known vulnerabilities :

- These components can be defined as the third-party apps or software or platforms that are outdated and contain bugs that are public to all, that is- sites like https://www.exploit-db.com contain the full detail as to how to exploit the bugs to put the security of the whole website under severe threat.
- This vulnerability arises with the fact that a website finds it difficult to code everything while making its website functional like -transaction, location, chats, etc.
- So, in order to ease the process of building the website, many websites use third-party apps which do the tasks. But the main problem is that those apps can be harmful to their system if they are not updated regularly. Components with known vulnerabilities are considered to be one of the top 10 web application vulnerabilities listed by OWASP.
- Many websites security gets compromised when they use components having known vulnerabilities.

How to spot:
There are automated tools available online ex- drop scanner which display all the outdated components present in the software. These automated tools save time for security experts by directly pointing out flaws of the website.

Business impact:
When a website uses such outdated software, the hacker can misuse the flaws and get inside the database of the website and can cause serious damage to the website by stealing critical information.

UN validated Redirects and Forwards

UN validated Redirects and Forward Vulnerability, also sometimes referred to as URL Redirection Vulnerability, is a type of bug found in the Web Application. In this type of vulnerability, the attacker uses to manipulate the URL and sends it to the victim. As soon as the victim opens the URL, the website redirects it to a malicious website or website to which the attacker wants the user to get redirected. The attacker generally uses to exploit this type of Vulnerability with the help of manual manipulation in the URL or with the help of several tools like *Burpsuite*, which gives an attacker several types of ways due to which he can manipulate the URL to get Redirected.

10) What are the roles and responsibilities of SDL?

A) OWASP provides a secure coding practices checklist that includes 14 areas to consider in your software development life cycle. Of those secure coding practices, we're going to focus on the top eight secure programming best practices to help you protect against vulnerabilities.

1. Security by Design
2. Password Management
3. Access Control
4. Error Handling and Logging
5. System Configuration
6. Threat Modeling
7. Cryptographic Practices
8. Input Validation and Output Encoding

Security by Design

Security needs to be a priority as you develop code, not an afterthought. Organizations may have competing priorities where software engineering and coding are concerned. Following software security best practices can conflict with optimizing for development speed. However, a "security by design" approach that puts security first tends to pay off in the long run, reducing the future cost of technical debt and risk mitigation. An analysis of your source code should be conducted throughout your software development life cycle (SDLC), and security automation should be implemented.

Password Management

Passwords are a weak point in many software systems, which is why multi-factor authentication has become so widespread. Nevertheless, passwords are the most common security credential, and following secure coding practices limits risk. You should require all passwords to be of adequate length and complexity to withstand any typical or common attacks. OWASP suggests several coding best practices for passwords, including:

- Disable password entry after multiple incorrect login attempts.

We have also written about password expiration policies and whether they are a security best practice in a modern business environment.

Access Control

Take a "default deny" approach to sensitive data. Limit privileges and restrict access to secure data to only users who need it. Deny access to any user that cannot demonstrate authorization. Ensure that requests for sensitive information are checked to verify that the user is authorized to access it.

Learn more about access controls for remote employees and cloud access management.

Error Handling and Logging

Software errors are often indicative of bugs, many of which cause vulnerabilities. Error handling and logging are two of the most useful techniques for minimizing their impact. Error handling attempts to catch errors in the code before they result in a catastrophic failure. Logging documents errors so that developers can diagnose and mitigate their cause.

System Configuration

Clear your system of any unnecessary components and ensure all working software is updated with current versions and patches. If you work in multiple environments, make sure you're managing your development and production environments securely.

Outdated software is a major source of vulnerabilities and security breaches. Software updates include patches that fix vulnerabilities, making regular updates one of the most vital, secure coding practices.

Threat Modeling

Document, locate, address, and validate are the four steps to threat modeling. To securely code, you need to examine your software for areas susceptible to increased threats of attack. Threat modeling is a multi-stage process that should be integrated into the software lifecycle from development, testing, and production.

UNIT-III

Secure coding practices and OWASP Top 10

PART-A

1) What is programmatic security and declarative security?

A) programmatic security:

Programmatic security involves an EJB component or servlet using method calls to the security API, as specified by the Java EE security model, to make business logic decisions based on the caller or remote user's security role. Programmatic security should only be used when declarative security alone is insufficient to meet the application's security model.

The API for programmatic security consists of methods of the Java EE Security API `Security Context` Interface, and methods of the `EJB Context` Interface and the servlet (`HttpServletRequest`) interface. The Glassfish Server supports these interfaces as specified in the Java EE specification.

Declarative Security

Declarative security means that the security mechanism for an application is declared and handled externally to the application. Deployment descriptors describe the Java EE application's security structure, including security roles, access control, and authentication requirements.

The Glassfish Server supports the deployment descriptors specified by Java EE and has additional security elements included in its own deployment descriptors. Declarative security is the application deployer's responsibility. For more information about Glassfish Server deployment descriptors, see the Glassfish Server Open Source Edition Application Deployment Guide.

There are two levels of declarative security, as follows:

- Application Level Security
- Component Level Security

Application Level Security

For an application, roles used by any application must be defined in `@DeclareRoles` annotations in the code or `role-name` elements in the application deployment descriptor (`application.xml`).

By default, group principal names are mapped to roles of the same name. Accordingly, the Default Principal To Role Mapping setting is enabled by default on the Security page of the Glassfish Server Administration Console..

Component Level Security

Component level security encompasses web components and EJB components.

A secure web container authenticates users and authorizes access to a servlet or JSP by using the security policy laid out in the servlet XML deployment descriptors

2) What are the techniques that can be used for input validation and sanitization?

validation checks whether an input — say on a web form — complies with specific policies and constraints (for example, single quotation marks). For example, consider thefollowing input:

```
<input id="num" name="num" type="number" />
```

If there's no validation, nothing prevents an attacker from exploiting the form by enteringunexpected inputs instead of an expected number. He or she could also try to execute code directly if submitted forms are stored in a database, which is pretty common.

To prevent such a bad situation, developers must add a validation step where the data is inspected before proceeding. For example, using a popular language like PHP, you cancheck the data type, the length, and many other criteria.

Sanitizing consists of removing any unsafe characters from user inputs, and validating will check to see if the data is in the expected format and type. Sanitizing modifies the input to ensure it's in a valid format for display, or before insertion in a database.

The biggest problem with sanitization is the false impression of security it might give. Stripping unwanted chars and HTML tags is only one layer of checking. It's often poorly executed and removes too much information like legitimate quotes and special chars while it does not cover all angles of attack. You cannot apply generic rules blindly.

The context is the key, which includes the programming languages in use. More on this later, but it's important to follow a principle called "escape late" (for example, just before output) because you know the exact context where the data is used.

3) How do logging and auditing differ in Secure Coding?

A) The difference between audit logs and regular system logs (e.g., error logs, operational logs, etc.) is the information they contain, their purpose, and their immutability. Whereas regular system logs are designed to help developers troubleshoot errors, audit logs help organizations document a historical record of activity for compliance purposes and other business policy enforcement. A log from any network device, application, host, or operating system can be classified as an audit log if it contains the information mentioned above and is used for auditing purposes. Compliance frameworks also generally require organizations to meet long-term retention policies, which is why audit logs aim to be immutable so that no user or service can alter audit trails.

Administrative activity

This includes events like creating or deleting a user account, such as deleting a user from your CRM tool (e.g., Salesforce).

Data access and modification

This includes events where a user views, creates, or modifies data, such as downloading a file from payroll software (e.g., Workday).

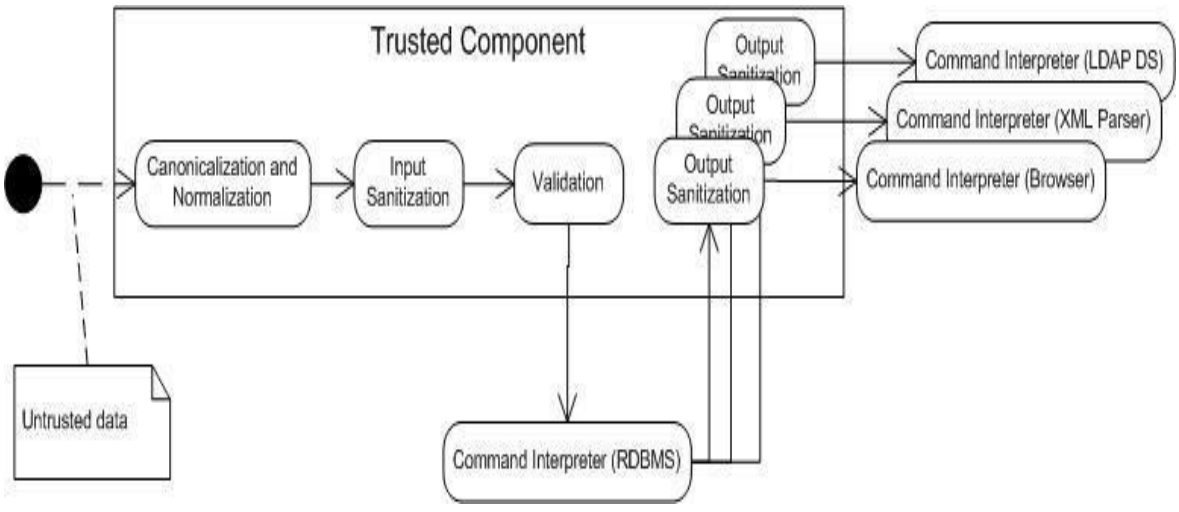
Most technologies in your tech stack will offer a UI where you can enable audit log collection. Depending on the specific tool, you may also have more granular control over audit log collection. For example, cloud vendors such as Amazon Web Services, Microsoft Azure, and Google Cloud automatically collect a wide range of audit logs. However, you may have to enable audit logging for certain services or certain types of activity to ensure you have enough data to prove compliance or investigate an incident.

4) What is input and output sanitization in Secure Coding?

A Java program can contain both internally developed and third-party code. Java was designed to allow the execution of un trusted code; consequently, third-party code can operate in its own trusted domain. The public API of such third-party code can be considered to be a trust boundary. Data that crosses a trust boundary should be validated unless the code that produces this data provides guarantees of validity. A subscriber or client may omit validation when the data flowing into its trust boundary is appropriate for use as is. In all other cases, inbound data must be validated.

Injection Attacks

Data received by a component from a source outside the component's trust boundary can be malicious and can result in an injection attack, as shown in the scenario in Figure 1.



Programs must take steps to ensure that data received across a trust boundary is appropriate and not malicious. These steps can include the following:

Validation: Validation is the process of ensuring that input data falls within the expected domain of valid program input. This requires that inputs conform to type and numeric range requirements as well as to input invariants for the class or subsystem.

Sanitization: In many cases, the data is passed directly to a component in a different trusted domain. Data sanitization is the process of ensuring that data conforms to the requirements of the subsystem to which it is passed. Sanitization also involves ensuring that data conforms to security-related requirements regarding leaking or exposure of sensitive data when output across a trust boundary. Sanitization may include the elimination of unwanted characters from the input by means of removing, replacing, encoding, or escaping the characters. Sanitization may occur following input (input sanitization) or before the data is passed across a trust boundary (output sanitization). Data sanitization and input validation may coexist and complement each other. Many command interpreters and parsers provide their own sanitization and validation methods. When available, their use is preferred over custom sanitization techniques because custom-developed sanitization can often neglect special cases or hidden complexities in the parser. Another problem with custom sanitization code is that it may not be adequately maintained when new capabilities are added to the command interpreter or parser software.

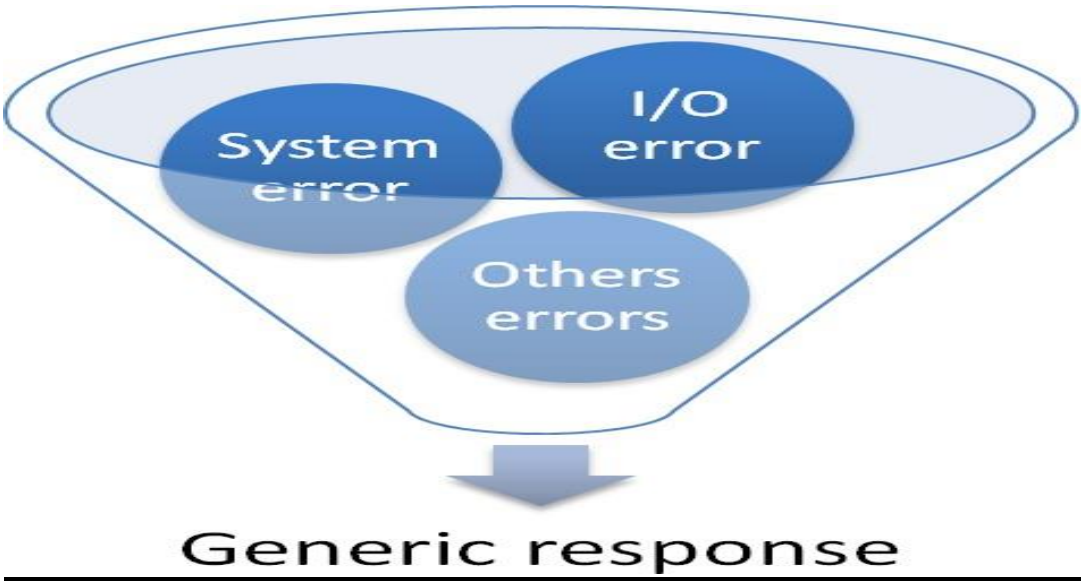
5) Why is error handling important security?

Error handling is a part of the overall security of an application. Except in movies, an attack always begins with a Reconnaissance phase in which the attacker will try to gather as much technical information (often name and version properties) as possible about the target, such as the application server, frameworks, libraries, etc.

Unhandled errors can assist an attacker in this initial phase, which is very important for the rest of the attack.

The article shows how to configure a global error handler as part of your application's runtime configuration. In some cases, it may be more efficient to define this error handler as part of your code. The outcome being that when an unexpected error occurs then a generic response is returned by the application but the error details are logged server side for investigation, and not returned to the user.

The following schema shows the target approach:



As most recent application topologies are API based, we assume in this article that the backend exposes only a REST API and does not contain any user interface content. The application should try and exhaustively cover all possible failure modes and use 5xx errors only to indicate responses to requests that it cannot fulfill, but not provide any content as part of the response that would reveal implementation details. For that, RFC 7807 - Problem Details for HTTP APIs defines a document format.

For the error logging operation itself, the logging cheat sheet should be used. This article focuses on the error handling part.