DEPARTMENT OF COMPUTER SCIENCE

I M.Sc CS – SECURITY IN COMPUTING

UNIT – I INTRODUCTION AND BASIC CONCEPTS:

Threats, vulnerabilities, controls; risk; Breaches; confidentiality, integrity, availability; Attacks, Exploits. Information Gathering (Social Engineering, Foot Printing & Scanning). Open Source/ Free/ Trial Tools: nmap, zenmap, Port Scanners, Network scanners.

OVERVIEW OF COMPUTER SECURITY

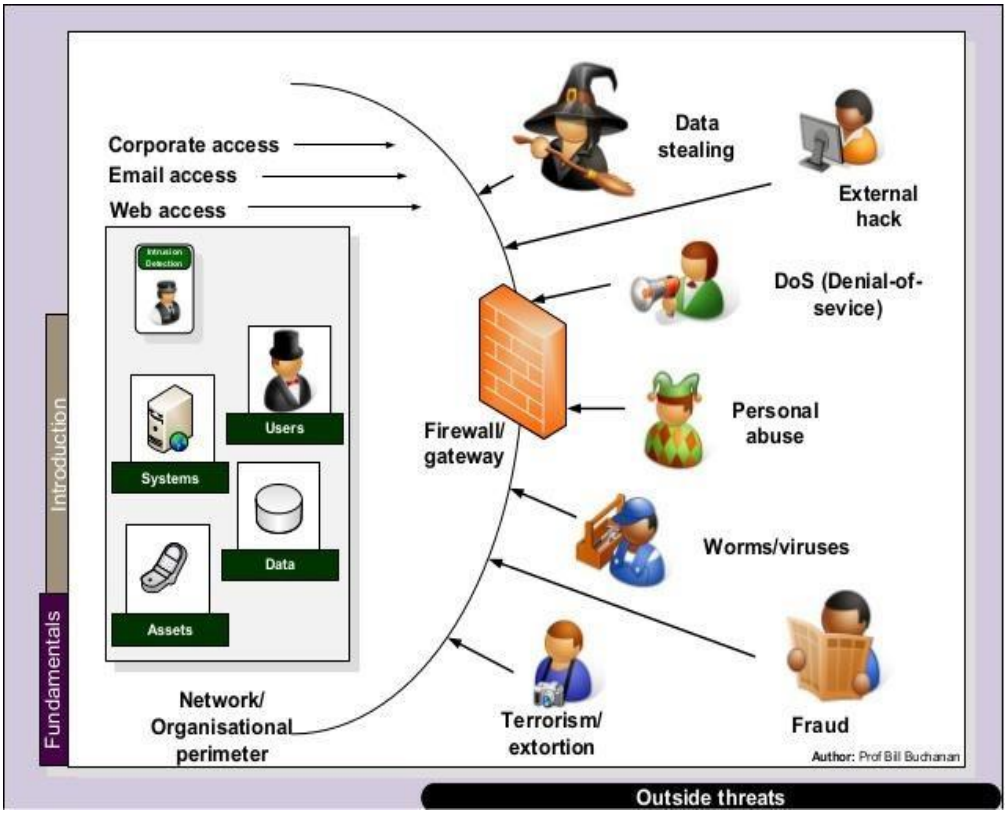
- A **computing system**: is a collection of hardware, software, data, and people that an organization uses to do computing tasks
- Computer security means protect our computing

system Main aspects are:

- Prevention:- Prevent your assets from being damaged
- Detection :- Detect when assets has been damage
- Reaction:- Recover your assets

Computer Security: - Ensuring the data stored in a computer cannot be read or compromised by an individual‘s without authorization.

- Most computer security measures involve data encryption and passwords.
- The purpose of computer security is to device ways to prevent the weaknesses from being



SECURITY CONCEPTS

Three Goals in Computing Security

Three goals of computer security are

1. Confidentiality
2. Integrity
3. Availability

1.Confidentiality:

It ensures that computer-related assets are accessed only by authorized parties. Confidentiality is sometimes called secrecy or privacy.

-Difficult to ensure

- Easy to assess

- Confidentiality is the ability to hide information from those people unauthorized to view it. It is perhaps the most obvious aspect of the CIA(Confidentiality, Integrity and Availability) triad when it comes to security; but correspondingly, it is also the one which is attacked most often.
- Cryptography and Encryption methods are an example of an attempt to ensure confidentiality of data transferred from one computer to another.
- A good example of methods used to ensure confidentiality is an account number or routing number when banking online.
- Data encryption is a common method of ensuring confidentiality. User IDs and passwords constitute a standard procedure; two-factor authentication is becoming the norm.
- Other options include biometric verification and security tokens, key fobs or soft tokens.
- In addition, users can take precautions to minimize the number of places where the information appears and the number of times it is actually transmitted to complete a required transaction.

Different approaches for achieving confidentiality are

- **Access control**: - specify who can access. One access control mechanism for preserving confidentiality is cryptography
- **Identification and Authentication**

Two concepts in confidentiality are

1. **Data Confidentiality**: - assures that confidential information is not disclosed to authorize individuals.

- Only the people who are authorized to do so can gain access to sensitive data. Imagine your bank records.
- You should be able to access them, of course, and employees at the bank who are helping you with a transaction should be able to access them, but no one else should.

2. **Privacy:** The right of individuals to hold information about themselves in secret, free from the knowledge of others

2.Integrity:

It means that assets can be modified only by authorized parties or only in authorized ways.

- Much difficult to measure Two concepts in integrity are

1. **Data Integrity**:- Information and programs are changed only in authorized manner
2. **System Integrity**: - System performs its operation in unimpaired manner that means state of the system not changed.

Integrity mechanisms fall into two classes:

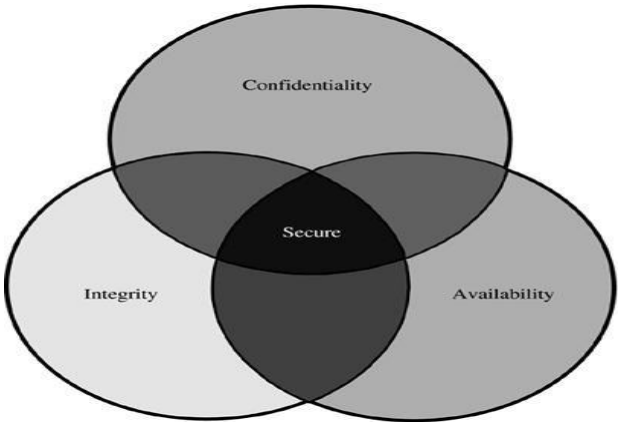
***Prevention* mechanisms and *detection* mechanisms**.

Prevention mechanisms seek to maintain the integrity of the data by blocking any unauthorized attempts to change the data or any attempts to change the data in unauthorized ways.

Detection mechanisms do not try to prevent violations of integrity; they simply report that the data's integrity is no longer trustworthy. The mechanisms may report the actual cause of the integrity violation (a specific part of a file was altered), or they may simply report that the file is now corrupt.

3.Availability: it means that assets are accessible to authorized users in all time

- Availability applies both to data and to service.
- Failure to this goal (availability)is known as Denial of service.
- Availability is an important aspect of reliability as well as of system design because an unavailable system is at least as bad as no system at all



One of the challenges in building a secure system is finding the right balance among the goals, which often conflict.

Along with three objectives system should also ensure

1. **Authentication**: Computer system be able to verify identity of user.

   Authentication technology provides access control for systems by checking to see if a user's credentials match the credentials in a database of authorized users or in a data authentication server.

   Users are usually identified with a user ID, and authentication is accomplished when the user provides a credential, for example a password, that matches with that user ID.

2. **Accountability**: Every individual who works with an information system should have specific responsibilities for information assurance.
3. **Non repudiation**: non-repudiation is the assurance that someone cannot deny the validity of something. Non-repudiation is a legal concept that is widely used in information security and refers to a service, which provides proof of the origin of data and the integrity of the data. In other words, non-repudiation makes it very difficult to successfully deny who/where a message came from as well as the authenticity of that message.

   Digital signatures can offer non-repudiation when it comes to online transactions, where it is crucial to ensure that a party to a contract or a communication can't deny the authenticity of their signature on a document or sending the communication in the first place.

   In this context, non-repudiation refers to the ability to ensure that a party to a contract or a communication must accept the authenticity of their signature on a document or the sending of a message.

NEED OF SECURITY

Why is computer security important?

Computer security is important, primarily to keep your information protected. It‘s also important for your computer‘s overall health, helping to prevent viruses and malware and allowing programs to run more smoothly. Security is needed due to following reason

1. Privacy:- It defines the right of individuals to hold information about themselves in secret, free from the knowledge of others
2. Accuracy: - Most of damages of data is caused by errors and omissions. An organization always needs accurate data for transaction processing, providing better service and making
3. Threats by dishonest employ
4. Computer Crimes:- When computer resources can be misused for unauthorized or illegal function
5. Threats for fire and Natural Disasters:- fire and natural disasters like floods, storms, lightening etc

THREATS

- A threat to a computing system is a set of circumstances that has the potential to cause loss or harm.
- A threat can be an object or person or other entity that represents a constant danger to an asset
- There are many threats to a computer system, including **human-initiated and computer-initiated ones.**
- A threat is blocked by control of vulnerability (Weakness of the system).

We can view any threat as being one of four

•An **interception** means that some unauthorized party has gained access to an asset. The outside party can be a person, a program, or a computing system.

•In an **interruption** is an asset of the system becomes lost, unavailable, or unusable.

•If an unauthorized party not only accesses but tampers with an asset, is called as a **modification**.

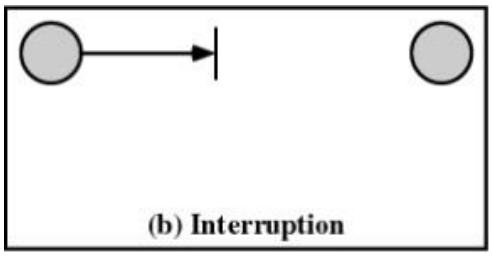
•An unauthorized party might create a **fabrication** of counterfeit objects on a computing system.

•The intruder may insert spurious transactions to a network communication system or add records to an existing database.

Kinds of threats

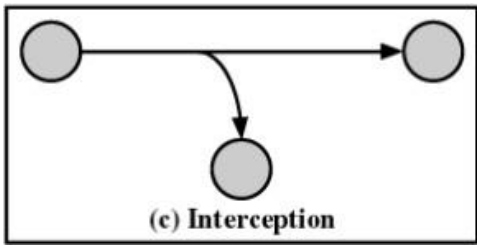
- **Interruption**

- An asset of the system is destroyed of becomes unavailable or unusable
  - Attack on availability
  - Destruction of hardware
  - Cutting of a communication line
  - Disabling the file management system



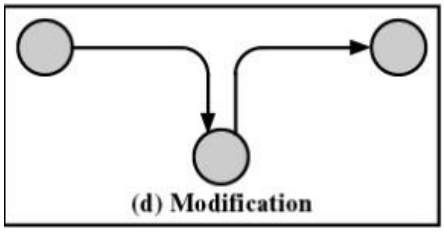
- Here the information being interrupted

- **Interception**
  - An unauthorized party gains access to an asset
  - Attack on confidentiality
  - Wiretapping to capture data in a network
  - Illicit copying of files or programs



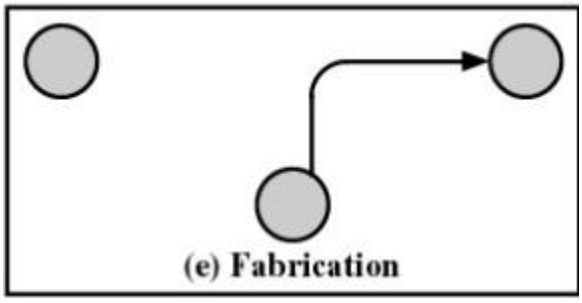
- There is a middleman or process or machine trying to intercept

- **Modification**
  - An unauthorized party not only gains access but tampers with an asset
  - Attack on integrity
  - Changing values in a data file
  - Altering a program so that it performs differently
  - Modifying the content of messages being transmitted in a network

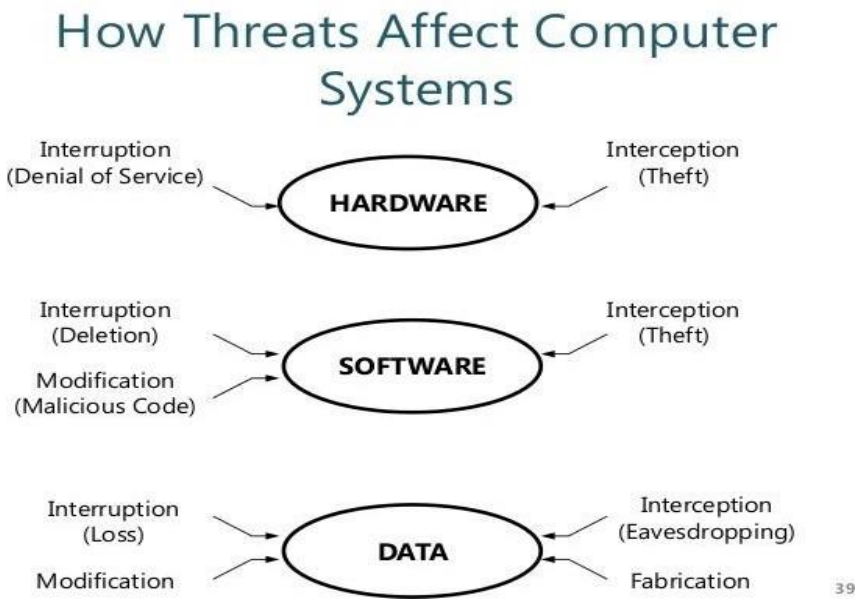


- Here middleman changes the data and send to the receiver

- **Fabrication**
  - An unauthorized party inserts counterfeit objects into the system
  - Attack on authenticity
  - Insertion of spurious messages in a network
  - Addition of records to a file



- Here sender not sends data to the receiver. Middleman fabricate the data



VULNERABILITY IN COMPUTER SECURITY

Vulnerability in computer security is a weakness in the design, implementation, management or operation of a system or information environment that can be exploited by a malicious source to threaten or directly violate your security and/or privacy policies.

These types of security breaches expose companies and individuals to serious risks, making it imperative to identify the vulnerabilities of each asset in order not to jeopardize the continuity of projects and prevent possible attacks.

Any external agent, such as a **hacker**, can cause attacks with varying degrees of damage to databases, software, information access systems, etc.

The result can be significant financial loss or a very negative impact on the organization's brand image.

Types of computer security vulnerabilities and examples

Vulnerability in computer security can be of **different types**, so knowing them can help you understand their importance.

Physical

This affects the availability of information and refers to security flaws in the environments in which data is stored and manipulated. This can include poor facilities, inadequate resources or lack of effective access control systems.

Natural or Environmental

This type refers to environmental conditions that can put information at risk. For example, elements such as moisture, dust, fire or flooding can adversely affect facilities, equipment, and infrastructure.

Hardware

The manufacturing defects of equipments, lack of updates or poor maintenance are factors that cause this type of vulnerability. Similarly, functional issues such as insufficient storage or speed can also compromise digital security.

Software

Software vulnerability in computer security is related to applications and programs with poor configurations and control systems. These vulnerabilities become gateways to computer systems or means of implementing malware.

Storage media

Any means of storing information can become vulnerability. Hard disks, pendrives and databases can be victims of attacks on the integrity, availability and confidentiality of data.

In ICTs

Vulnerabilities in ICTs (Information and Communication Technologies) concern information received or sent through all types of digital channels such as satellites, waves, wifi, optical fiber, etc.

Smartphones, tablets or computers can be the source of a security breach, underscoring the need to strengthen trusted internet connections.

Human

- People can cause vulnerabilities intentionally – fraud, phishing or impersonation, etc. or accidentally.
- The most common cause of computer security vulnerabilities among an organization's users is often a lack of cybersecurity training and awareness.
- This can lead to the use of unauthorized devices that may be infected, failure to use sufficiently secure passwords, clicking on dangerous links in emails, etc.
- Similarly, the application of digital hygiene measures, such as the use of protection tools like firewalls and antivirus, is essential.

CONTROL

Security Controls or cyber security controls are the most important factor used to develop the actions taken to prevent the organization's security risks. IT security controls are parameters implemented to protect the organization's data and assets.

Types of security control

1. Administrative Control: Administrative Control is a set of security rules, policies, procedures, or guidelines specified by the management to control access and usage of confidential information. It includes all the levels of employees in the organization and determines the privileged access to the resources to access data.

- User Management
- Privilege Management
- Employee Security, Clearance, and Evaluation
- Employee training and awareness, etc.

2. Physical Control: Physical Control is a set of IT security controls implemented physically to prevent unauthorized access to the data and security risks. Some examples of physical controls in cyber security controls are:

- Surveillance cameras
- Biometrics
- Identity Cards
- Alarm systems, etc.

3. Technical Control: Technical Control is to control the access of confidential information over the network using technology. Technical functions are involved in managing and controlling the access of the employee. Some examples of technical controls are:

- Access controls
- Firewalls
- Network Authentication
- Encryption, etc.

What Are The Goals Of Security Controls?

The overall purpose of implementing security controls is to reduce risks in an organization.
In other words, the primary goal of implementing security controls is to prevent or reduce the impact of a security incident.
The effective implementation of security control is based on its classification in relation to the security incident.

The common classifications types are listed below along with their corresponding description:

***Preventive Controls*:** These controls proactively provide protection by obstructing the initiation of attacks. Access Control Lists (ACLs) and firewalls are deployed to prevent unauthorized access, while anti-malware software plays a crucial role in detecting and neutralizing malicious software. Furthermore, administrative controls, such as directives and Standard Operating Procedures (SOPs), are implemented within organizations to establish and enforce security policies.

***Detective Controls*:** These controls are instrumental in identifying and alerting abnormal activities during or after an attack. Log records are a quintessential example of detective controls, enabling monitoring and tracking of security incidents. The critical significance of detective controls lies in their ability to facilitate timely intervention.

***Corrective Controls*:** Activated post-attack, these controls aim to alleviate or completely eradicate the inflicted damages. Backup systems support this process by restoring compromised data, while patch management systems address and rectify vulnerabilities exploited during the attack.

***Physical Controls*:** These controls ensure the physical security of premises and hardware, utilizing tools such as security cameras, alarms, and security personnel to deter and detect unauthorized access.

***Deterrent Controls*:** Serving to psychologically dissuade potential attackers, these controls aim to reduce the likelihood of attack attempts. They often manifest in the form of signage warning of legal repercussions for unauthorized entry or intrusion.

***Compensatory Controls*:** These controls are substitutes for primary control mechanisms, providing equivalent or superior protection through alternative methodologies or technologies. They are recommended by security standards, offering flexibility and resilience in the security posture.

RISK

A computer risk is anything that can harm a user information on a computer. This information can vary, in value, from computer to computer.
A hacker can steal information with a variety of methods such as exploiting the operating system and coding viruses or worms.
A user can protect their computer by implementing an antivirus and safely browsing the web.

Risks can be defined with this simple formula - **Risk = Threat + Vulnerability**.
Risks are generally determined by examining the threat actor and type of vulnerabilities that the system has.

Types of Risks

There are two types of cyber risks, which are as follows:

1. **External-** External cyber risks are those which come from outside an organization, such as cyberattacks, phishing, ransomware, DDoS attacks, etc.
2. **Internal-** Internal cyber risks come from insiders. These insiders could have malicious intent or are just not be properly trained.

Here are the key components of risk in computer security:

1. Threats

A threat is any potential event or action that could cause harm to a computer system. Common threats include:

- **Malware (viruses, worms, ransomware)**: Malicious software that can damage or exploit systems.
- **Phishing attacks**: Attempts to steal sensitive information via deceptive emails or websites.
- **Insider threats**: Employees or insiders who misuse their access to cause harm.
- **Denial of Service (DoS) attacks**: Attacks aimed at disrupting normal network service.
- **Advanced Persistent Threats (APT)**: Prolonged and targeted cyber-attacks often performed by nation-states.

2. Vulnerabilities

Vulnerabilities are weaknesses in a system that can be exploited by threats. These can include:

- **Software bugs**: Flaws or errors in code that attackers can exploit.
- **Misconfigured systems**: Poorly configured security settings that expose sensitive data or systems.
- **Unpatched systems**: Failing to apply updates or security patches leaves systems open to known exploits.
- **Weak passwords**: Poor password policies that make it easy for attackers to gain unauthorized access.

3. Likelihood

The likelihood refers to the probability that a specific vulnerability will be exploited by a threat. Some factors affecting likelihood include:

- **Attractiveness of the target**: Systems that handle valuable data are more likely to be targeted.
- **Known vulnerabilities**: If vulnerabilities are widely known, they are more likely to be exploited.
- **Presence of threat actors**: If a system operates in an environment with many skilled attackers, the likelihood of attack increases.

4. Impact

Impact refers to the consequences or damage that would occur if a vulnerability is successfully exploited. The impact can include:

- **Data loss or theft**: Sensitive data could be stolen, leading to financial loss or reputational damage.
- **System downtime**: Critical systems could be disabled, leading to productivity losses or service disruption.
- **Financial loss**: Costs of responding to an attack, such as regulatory fines, legal costs, or lost revenue.
- **Reputational damage**: Damage to the organization's trustworthiness, leading to loss of customers or partners.

5. Risk Mitigation

Risk mitigation involves taking steps to reduce the likelihood or impact of a threat. Common risk mitigation strategies include:

- **Encryption**: Protecting sensitive data by encoding it so that only authorized users can access it.
- **Firewalls and intrusion detection systems**: Monitoring network traffic to detect and prevent unauthorized access.
- **Regular updates and patching**: Ensuring that systems are updated to close known vulnerabilities.
- **Security policies and awareness**: Educating employees about best security practices to reduce human error.
- **Backup and recovery**: Having systems in place to restore data and systems in case of an attack.

6. Risk Assessment

Risk assessment is the process of identifying, evaluating, and prioritizing risks to help an organization decide where to allocate resources. It often involves:

- **Identifying threats and vulnerabilities**.
- **Calculating likelihood and impact**.
- **Prioritizing risks based on severity**.
- **Developing a risk mitigation plan**.

Organizations manage risk through proactive measures, such as security policies, technological defenses, and continuous monitoring.

BREACHES IN COMPUTER SECURITY

Breaches in computer security, often referred to as **data breaches** or **security breaches**, occur when an unauthorized party gains access to a system or network, resulting in the exposure, theft, or loss of sensitive information. These breaches can have severe consequences, including financial losses, legal liabilities, and damage to a company's reputation. Below are key aspects of security breaches:

1. Types of Security Breaches

a. Data Breaches

- A **data breach** happens when sensitive, confidential, or protected information is accessed or disclosed without authorization. This is often the result of hacking, malware, or even insider misuse.
- Commonly targeted data includes **personal identifiable information (PII)** such as Social Security numbers, financial data, medical records, and passwords.

b. Network Breaches

- In a **network breach**, an attacker gains unauthorized access to an organization’s internal network, often allowing them to move laterally to access various systems or data.
- Methods include exploiting vulnerabilities, **brute force attacks**, or using stolen credentials.

c. Ransomware Attacks

- In these attacks, malicious software encrypts the victim's files or systems, making them inaccessible. The attacker demands a ransom for the decryption key.
- Ransomware attacks often start with a phishing email or an exploit of a system vulnerability.

d. Denial of Service (DoS) and Distributed Denial of Service (DDoS)

- These attacks aim to overwhelm a system or network with excessive requests, causing it to become unavailable to users.
- While they do not typically involve data theft, they can still result in significant financial loss due to downtime.

e. Credential Stuffing

- Attackers use stolen username/password pairs from one breach to try and log into other accounts, often through automated login attempts.
- This takes advantage of the common user practice of reusing passwords across different services.

f. Insider Threats

- Security breaches can also result from **insider threats**, where employees, contractors, or other individuals with access intentionally or unintentionally compromise security. This can include stealing data, installing malware, or mishandling sensitive information.

2. Common Causes of Security Breaches

a. Phishing and Social Engineering

- **Phishing** attacks trick users into giving away sensitive information, such as login credentials or financial data. Attackers typically disguise themselves as legitimate entities via email, SMS, or even phone calls.
- **Spear phishing** is a more targeted version, often aimed at specific individuals within an organization.

b. Malware

- **Malware** such as viruses, worms, trojans, and spyware are common tools used by attackers to infiltrate systems and extract data or cause damage.
- Malware can be delivered via email attachments, downloads from compromised websites, or infected USB drives.

c. Unpatched Software

- Failing to apply **security patches** leaves systems vulnerable to known exploits. Attackers often use these known vulnerabilities as entry points for breaching systems.

d. Weak Passwords

- Poor password practices, such as using easily guessable or reused passwords, make systems vulnerable to brute-force attacks and credential stuffing.

e. Third-Party Vendors

- Organizations often rely on third-party vendors for services or software, and these vendors can become a weak link in the security chain. A breach in a vendor's system can lead to indirect access to an organization's data.

f. Insecure Cloud Storage

- Misconfigurations or lack of security controls in cloud environments, such as leaving sensitive data publicly accessible, can lead to data exposure.

3. Consequences of Security Breaches

a. Financial Loss

- Direct financial losses can result from ransom payments, fraud, or recovery efforts. There are also costs associated with legal fees, regulatory fines, and compensation to affected customers.

b. Reputational Damage

- A security breach can erode the trust of customers, partners, and stakeholders. Negative media attention can also impact the company's brand and lead to customer loss.

c. Legal and Regulatory Penalties

- Many industries are subject to strict data protection regulations, such as **GDPR** in Europe or **HIPAA** in healthcare. Failure to protect customer data can lead to hefty fines and legal action.

d. Operational Disruption

- Breaches can cause operational downtime as systems are shut down or quarantined to contain the threat. This disruption can lead to significant productivity losses.

e. Data Loss or Theft

- The loss or theft of sensitive data, such as intellectual property, trade secrets, or personal information, can have long-lasting negative effects on an organization.

4. Famous Examples of Security Breaches

a. Equifax Breach (2017)

- Personal information of over 147 million people, including Social Security numbers and credit card information, was exposed due to a vulnerability in Equifax's web application software.

b. Yahoo Breach (2013-2014)

- All 3 billion Yahoo user accounts were compromised in what is considered one of the largest data breaches in history. The stolen data included names, email addresses, and passwords.

5. Preventing and Responding to Security Breaches

a. Prevention Measures

- **Strong access controls**: Implement multi-factor authentication (MFA) and enforce strong password policies.
- **Regular updates and patches**: Ensure all systems, including third-party applications, are updated with the latest security patches.
- **Employee training**: Conduct regular security awareness training to help employees recognize phishing attempts and other social engineering tactics.
- **Encryption**: Encrypt sensitive data both in transit and at rest to protect it in case of unauthorized access.
- **Network monitoring and logging**: Continuously monitor network activity to detect and respond to suspicious behavior quickly.

CIA TRIAD

The three letters in "CIA triad" stand for Confidentiality, Integrity, and Availability. The CIA triad is a common model that forms the basis for the development of security systems. They are used for finding vulnerabilities and methods for creating solutions.

The confidentiality, integrity, and availability of information is crucial to the operation of a business, and the CIA triad segments these three ideas into separate focal points. This differentiation is helpful because it helps guide security teams as they pinpoint the different ways in which they can address each concern.

Ideally, when all three standards have been met, the security profile of the organization is stronger and better equipped to handle threat incidents.

1. Confidentiality

Confidentiality involves the efforts of an organization to make sure data is kept secret or private. To accomplish this, access to information must be controlled to prevent the unauthorized sharing of data—whether intentional or accidental. A key component of maintaining confidentiality is making sure that people without proper authorization are prevented from accessing assets important to your business. Conversely, an effective system also ensures that those who need to have access have the necessary privileges.

For example, those who work with an organization’s finances should be able to access the spreadsheets, bank accounts, and other information related to the flow of money. However, the vast majority of other employees—and perhaps even certain executives—may not be granted access. To ensure these policies are followed, stringent restrictions have to be in place to limit who can see what.

There are several ways confidentiality can be compromised. This may involve direct attacks aimed at gaining access to systems the attacker does not have the rights to see. It can also involve an attacker making a direct attempt to infiltrate an application or database so they can take data or alter it.

These direct attacks may use techniques such as **man-in-the-middle (MITM) attacks**, where an attacker positions themselves in the stream of information to intercept data and then either steal or alter it. Some attackers engage in other types of network spying to gain access to credentials. In some cases, the attacker will try to gain more system privileges to obtain the next level of clearance.

However, not all violations of confidentiality are intentional. Human error or insufficient security controls may be to blame as well. For example, someone may fail to protect their password—either to a workstation or to log in to a restricted area. Users may share their credentials with someone else, or they may allow someone to see their login while they enter it. In other situations, a user may not properly encrypt a communication, allowing an attacker to intercept their information. Also, a thief may steal hardware, whether an entire computer or a device used in the login process and use it to access confidential information.

To fight against confidentiality breaches, you can classify and label restricted data, enable access control policies, encrypt data, and use multi-factor authentication (MFA) systems. It is also advisable to ensure that all in the organization have the training and knowledge they need to recognize the dangers and avoid them.

2. Integrity

Integrity involves making sure your data is trustworthy and free from tampering. The integrity of your data is maintained only if the data is authentic, accurate, and reliable.

For example, if your company provides information about senior managers on your website, this information needs to have integrity. If it is inaccurate, those visiting the website for information may feel your organization is not trustworthy. Someone with a vested interest in damaging the reputation of your organization may try to hack your website and alter the descriptions, photographs, or titles of the executives to hurt their reputation or that of the company as a whole.

Compromising integrity is often done intentionally. An attacker may bypass an intrusion detection system (IDS), change file configurations to allow unauthorized access, or alter the logs kept by the system to hide the attack. Integrity may also be violated by accident. Someone may accidentally enter the wrong code or make another kind of careless mistake. Also, if the company's security policies, protections, and procedures are inadequate, integrity can be violated without any one person in the organization accountable for the blame.

To protect the integrity of your data, you can use hashing, encryption, digital certificates, or digital signatures. For websites, you can employ trustworthy certificate authorities (CAs) that verify the authenticity of your website so visitors know they are getting the site they intended to visit.

A method for verifying integrity is non-repudiation, which refers to when something cannot be repudiated or denied. For example, if employees in your company use digital signatures when sending emails, the fact that the email came from them cannot be denied. Also, the recipient cannot deny that they received the email from the sender.

3. Availability

Even if data is kept confidential and its integrity maintained, it is often useless unless it is available to those in the organization and the customers they serve. This means that systems, networks, and applications must be functioning as they should and when they should. Also, individuals with access to specific information must be able to consume it when they need to, and getting to the data should not take an inordinate amount of time.

If, for example, there is a power outage and there is no **disaster recovery system** in place to help users regain access to critical systems, availability will be compromised. Also, a natural disaster like a flood or even a severe snowstorm may prevent users from getting to the office, which can interrupt the availability of their workstations and other devices that provide business-critical information or applications. Availability can also be compromised through deliberate acts of sabotage, such as the use of **denial-of-service (DoS) attacks** or ransomware.

To ensure availability, organizations can use redundant networks, servers, and applications. These can be programmed to become available when the primary system has been disrupted or broken. You can also enhance availability by staying on top of upgrades to software packages and security systems. In this way, you make it less likely for an application to malfunction or for a relatively new threat to infiltrate your system. Backups and full disaster recovery plans also help a company regain availability soon after a negative event.

Should we Use the CIA Triad?

The CIA security triad is valuable in assessing what went wrong—and what worked—after a negative incident. For example, perhaps availability was compromised after a malware attack such as **ransomware**, but the systems in place were still able to maintain the confidentiality of important information. This data can be used to address weak points and replicate successful policies and implementations.

When should we use the CIA triad?

However, it is particularly helpful when developing systems around **data classification** and managing permissions and access privileges. You should also stringently employ the CIA triad when addressing the cyber vulnerabilities of your organization. It can be a powerful tool in disrupting the Cyber Kill Chain, which refers to the process of targeting and executing a cyberattack. The CIA security triad can help you hone in on what attackers may be after and then implement policies and tools to adequately protect those assets.

In addition, the CIA triad can be used when training employees regarding cybersecurity. You can use hypothetical scenarios or real-life case studies to help employees think in terms of the maintenance of confidentiality, integrity, and availability of information and systems.

ATTACKS

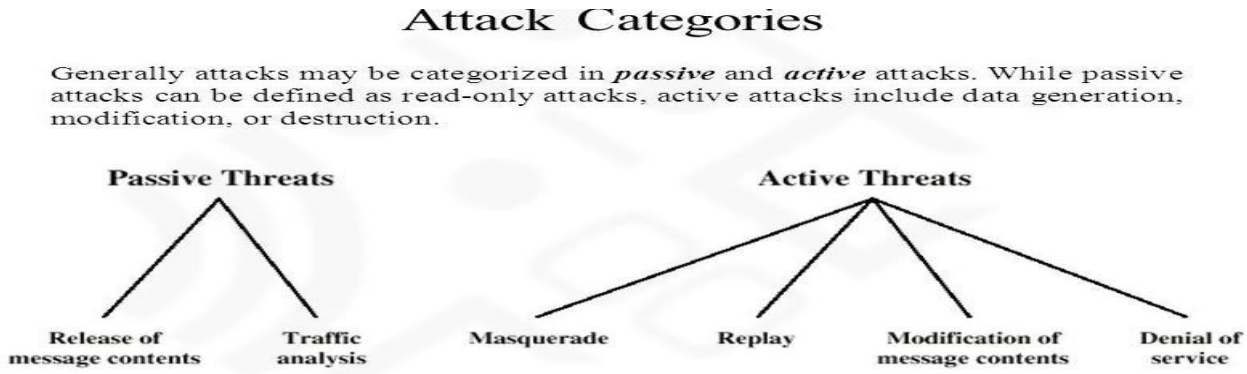
- Attack is the process of gaining the access of data by unauthorized user.
- It is an Act or attack that exploit vulnerability(Weakness of the system)

Definition - What does Attack mean?

An attack is an information security threat that involves an attempt to obtain, alter, destroy, remove, implant or reveal information without authorized access or permission. It happens to both individuals and organizations.

Two types of attacks are

1. Passive attack:-data just accessed by third party, no modification, does not affect system resources
2. Active attack:- data will be modified



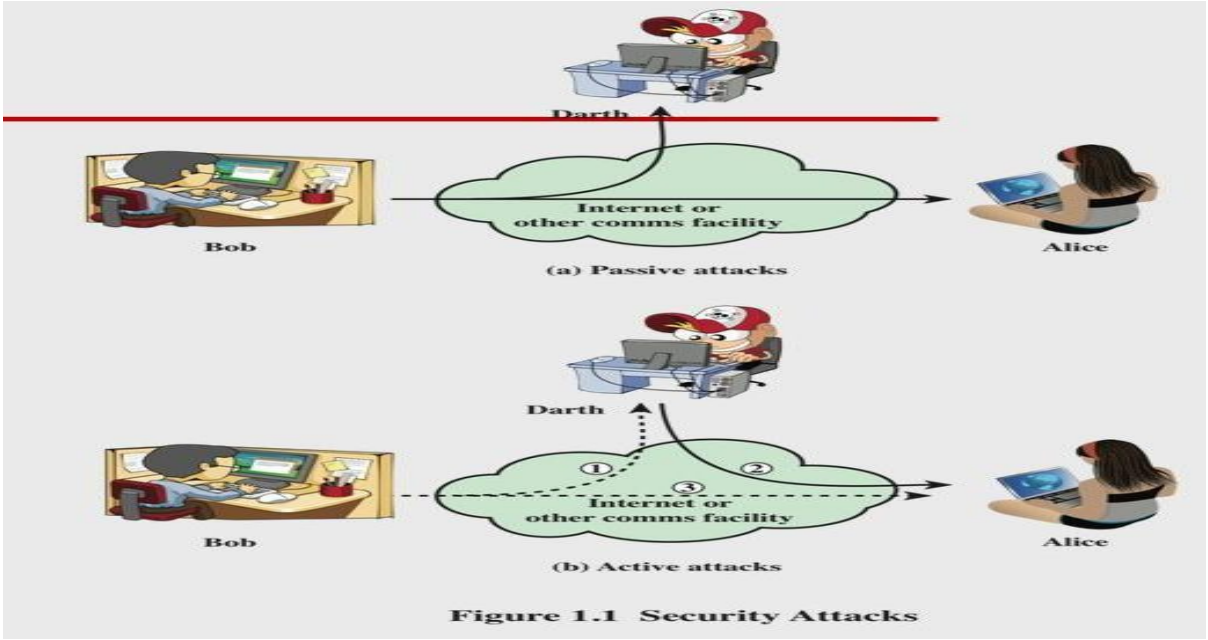
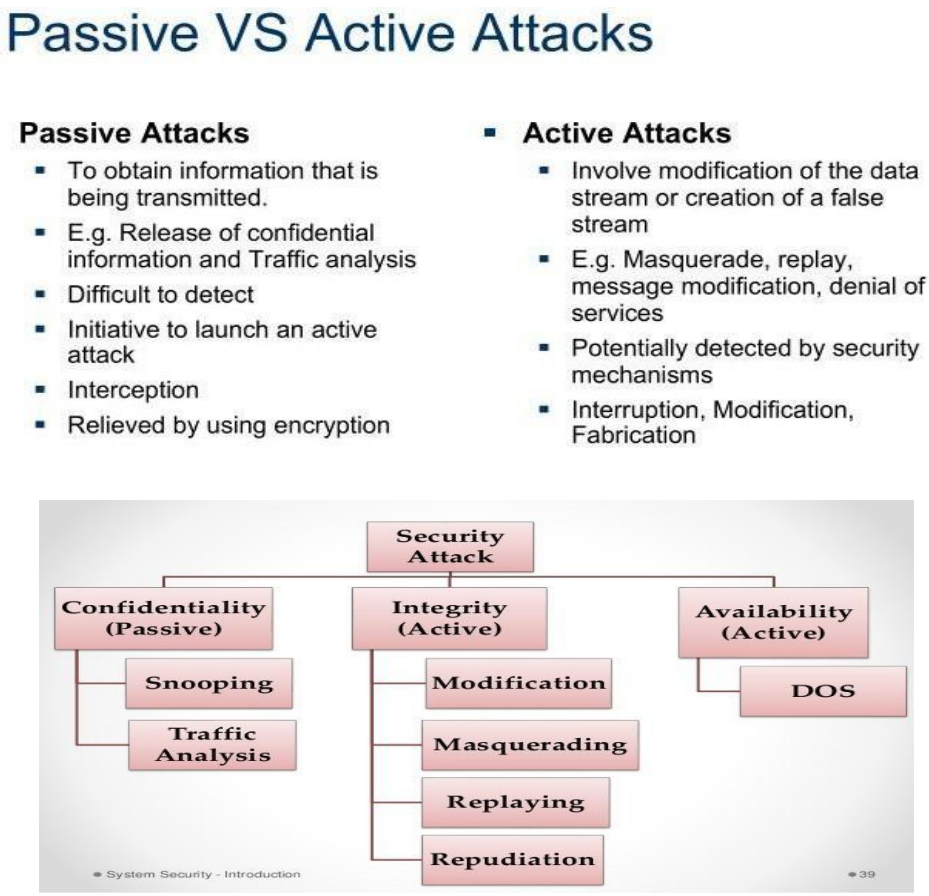


Figure 1.1 Security Attacks

- **Passive Attacks**

- **Release of message contents** for a telephone conversion, an electronic mail message, and a transferred file are subject to these threats
- **Traffic analysis:-** By analyzing the traffic flow between sender and receiver third party access the data

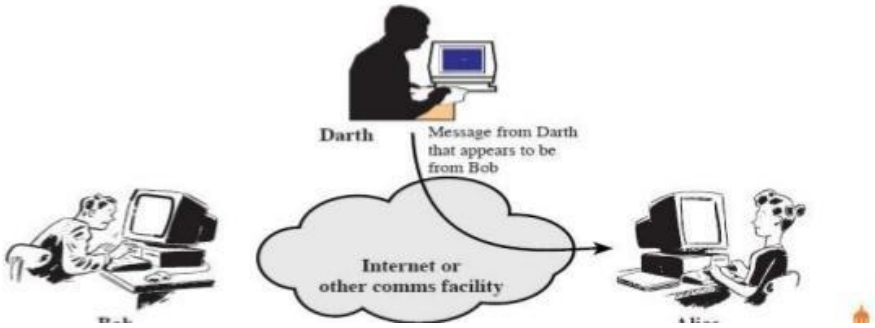
- **Active Attacks**
  - **Masquerade** takes place when one entity pretends to be a different entity
  - **Replay** involves the passive capture of a data unit and its subsequent retransmission to produce an unauthorized effect
  - **Modification** of messages means that some portion of a legitimate message is altered, or that messages are delayed or reordered, to produce an unauthorized effect
  - **Denial of service** prevents or inhibits the normal use or management of communications facilities
    - Disable network or overload it with messages



1. **Masquerade attack**

The third party sends the same message to the receiver and receiver receives it with the name of sender.

A masquerade is a type of attack where the attacker act as an authorized user system in order to gain access to it or to gain greater privileges than they are authorized for.

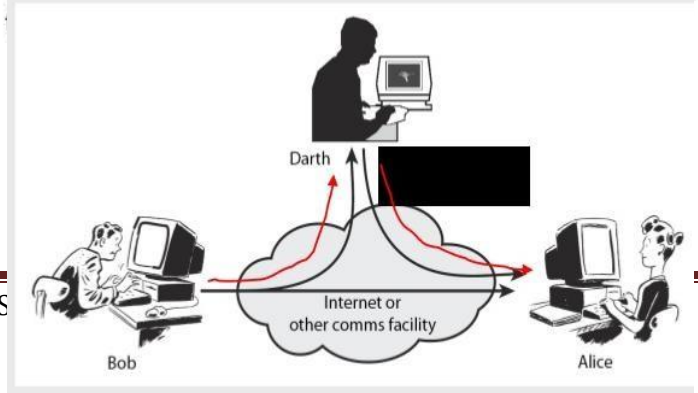


2. **Replay attack**

3. Here receiver receives two messages. One from sender and another from third party.
4. Receiver did not know which one is correct

3.Data Modification attack

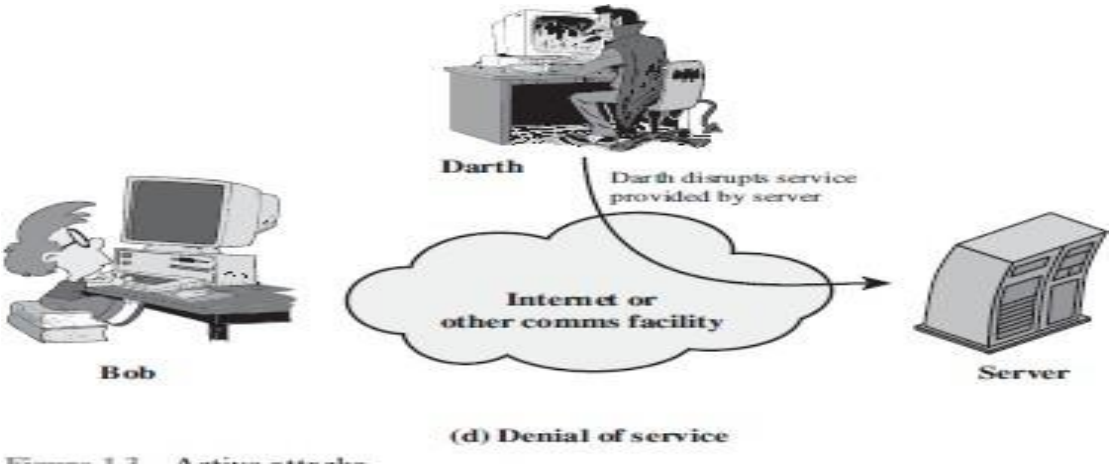
- Modification is integrity violation.
- An unauthorized party not only gains access to but tampers with an asset.
- This is an attack on the integrity.
- Examples include changing values in a data file, altering a program so that it performs differently, and modifying the content of a message being



4.Denial of Service

- A denial-of-service (DoS) is any type of attack where the attackers (hackers) attempt to prevent legitimate users from accessing the service.
- In a DoS attack, the attacker usually sends excessive messages asking the network or server to authenticate requests that have invalid return addresses.
- The network or server will not be able to find the return address of the attacker when sending the authentication approval, causing the server to wait before closing the connection.
- When the server closes the connection, the attacker sends more authentication messages with invalid return addresses. Hence, the process of authentication and server wait will begin again, keeping the network or server busy.

- Here the third party interrupts (disrupts) the services sends by the server.
- Disruption of entire network either by disabling the network or by overloading it with message , so as to degrade performance



(d) Denial of service

EXPLOIT

An exploit (in its noun form) is a segment of code or a program that maliciously takes advantage of vulnerabilities or security flaws in software or hardware to infiltrate and initiate a denial-of-service (**DoS) attack** or install malware, such as **spyware**, **ransomware**, **Trojan horses**, **worms**, or viruses. So the exploit is not the malware itself but is used to deliver the malware. To exploit (in its verb form) is to successfully carry out such an attack.

How Do Exploits Work?

When developers produce an operating system (OS) for a device, write code for software, or develop an application, bugs often appear due to inherent imperfections. These bugs can create a vulnerability in the

system, and an exploit searches out such vulnerabilities and looks for a way to exploit databases and networks or systems.

If the bug is not reported and "patched," it becomes an entryway, so to speak, for cyber criminals to conduct an exploit. With so many devices connected together in the modern world, as in the Internet of Things (IoT), for example, an exploit does not just compromise a singular device, but it can become a security vulnerability for a whole network.

Types, Groups, and Categories of Exploits

The Different Types of Exploits

Hardware

Hardware, to various degrees, must run on an OS, whether it be a complex OS for a PC or a simpler OS for an edge device. Vulnerabilities in the OS become entry points for an exploit, which can corrupt the memory or cause the device to "freeze."

Software

Software bugs, a normal consequence of software development, can become vulnerabilities open to exploits if not patched or fixed. Some of the common exploit methods include memory safety violations, input validation errors, side-channel attacks, and privilege confusion bugs.

Network

Each of the components of a network offers the possibility of vulnerability, whether hardware, software, or firewall configurations. Some attacks that may be part of an exploit can be domain hijacking, DoS and **distributed denial-of-service (DDoS) attacks**, and malware.

Personnel

Even personnel can be exploited. Cyber criminals may target their devices and credentials by means of **social engineering attacks**, **spear phishing**, and honey trapping. Training and access control are crucial to mitigating this vulnerability.

Physical Site

Exploits can be conducted on-site and if deficient physical security or inadequate access control exists. Just as a thief can break in and steal, a cyber criminal can break in (physically or remotely) and conduct an exploit that compromises an entire network.

Groups in Which Exploits Can Be Categorized

Zero-day exploits

This is a previously unknown exploit or an unknown opportunity for an exploit due to vulnerabilities. Anticipating **zero-day exploits** is crucial to developing patches or other strategies for mitigating the vulnerability or threat.

Known vulnerabilities

Known vulnerabilities have been identified and documented. Patches and other "fixes" can be issued, but cyber criminals can also get hold of the documentation and design an exploit. The main risk factor is that organizations often do not apply the patch or repair an issue quickly enough to eliminate a vulnerability.

How Do Exploits Occur?

Remote exploits

Remote exploits are run on an external computer, via an intranet or other network, exploiting a security vulnerability without prior access to the system. Its purpose is to either access or steal data or install malware to either a single computer or a complete system or network.

Local exploits

Local exploits can only be run if the malicious party has access to a machine on the network using a compromised account.

Client exploits

Client exploits influence or attack a user, misleading the user to click and download malware that can then compromise the network or system.

What Is an Exploit Kit?

Exploit kits silently and automatically seek to exploit any vulnerabilities identified on a user's machine when they are web browsing. They are largely automated in nature and have become the preferred method for the distribution of **remote access tools (RATs)** or mass malware by cyber criminals, especially those seeking to profit from an exploit.

Often, the goal is to gain control of devices in a simplified and automated manner. A sequence of events takes place within an exploit kit for the attack to be successful. It starts with a redirect to a landing page, followed by the execution of the exploit, and finally, the delivery of the payload, gaining control of the host.

Exploit kits can also be used in **penetration testing** to evaluate the security of the system. For example, the **Fortinet exploit kit** is used to run a simulation exercise on a system to detect vulnerabilities.

How To Recognize an Exploit Attack

Slow performance

There are multiple issues that can cause a machine or system to run slowly, and infection as the result of an exploit is one of them. So if you are used to seeing fast performance, and your device slows suddenly as if bogged down, it may be due to a malware infection.

Frequent crashes or freezes

Freezing, crashing, and the dreaded blue screen of death can all be caused by technical issues due to incompatibility between hardware and software, but malware infections can also be the cause.

Unexplained changed settings

Unusual behavior and changes you do not recall making, such as a changed default homepage in your browser, can be annoying, but they can be much more than annoying if caused by malicious software or unauthorized access.

Tons of pop-ups or ads where they should not be

Numerous pop-ups can disguise concealed malware threats, and annoying ads may actually be monitoring your browsing activity, hoping to collect data and passwords. Unsolicited emails and special offers may also be concealing similar intent.

Loss of storage space

Rapid, sudden loss of storage space can be the result of several underlying issues, but infection with malware is a primary reason and must be investigated before being eliminated as a possible cause.

How To Prevent an Exploit

It goes without saying that preventing exploits is preferable to fixing the damages. Certain strategies help prevent any component in the organization from being exploited.

1. **Software**: Apply patches and updates as soon as possible. Run antivirus software scans.
2. **Hardware**: Keep operating systems up-to-date. Scan with antivirus software, and institute control access protocols.
3. **Network**: Practice safe computing habits, control access, monitor the network for unusual activity, and **establish network security**.
4. **Personnel**: Train your employees in safe computing habits. Advise them on how to identify risks and prevent them. Enable multi-factor authentication (MFA) and other access control management strategies.
5. **Physical site**: Maintain good physical security, and monitor access.

INFORMATION GATHERING

Information gathering in computer security, also known as reconnaissance or footprinting, is a critical phase in the security assessment process. It involves collecting as much relevant data as possible about a target—such as a network, system, or organization—to identify potential vulnerabilities and plan attacks or defensive measures. Here's an overview of how information gathering is conducted and its role in security:

Types of Information Gathering

1. **Passive Information Gathering**:
    - **Publicly Available Information**: Gathering data from publicly accessible sources, such as websites, social media, company reports, and domain registration records.
    - **Search Engines**: Using search engines to find information about the target, including indexed web pages, documents, and past security incidents.

- **WHOIS Lookup**: Querying domain registration information to find details like the domain owner's contact information and domain registration dates.
- **Network Scanning**: Using tools to gather information about the network topology and IP addresses without directly interacting with the target systems.

2. **Active Information Gathering**:
   - **Port Scanning**: Identifying open ports and services on a target system using tools like Nmap. This helps in understanding what services are running and their versions.
   - **Vulnerability Scanning**: Scanning systems for known vulnerabilities using tools such as Nessus or OpenVAS to identify weaknesses that could be exploited.
   - **Social Engineering**: Directly interacting with individuals to obtain confidential information, such as posing as an internal IT support staff to gather system details or credentials.
   - **Network Mapping**: Analyzing the network structure to understand the relationships between different devices and systems within the network.
   - **OS Fingerprinting**: Identifying the operating system of a target machine to tailor attacks or defenses appropriately.

Tools and Techniques

- **Search Engines**: Google, Bing, and specialized search engines can be used to find information related to the target.
- **WHOIS Services**: Websites like WHOIS.net or domain registrars can provide domain registration details.
- **Network Scanners**: Tools like Nmap, Masscan, and Angry IP Scanner are used to discover devices and services on a network.
- **Vulnerability Scanners**: Tools like Nessus, Qualys, and OpenVAS scan systems for known vulnerabilities.
- **Social Engineering Tools**: Tools and techniques for crafting phishing emails or creating fake personas.
- **Reconnaissance Tools**: Tools like Maltego, theHarvester, and Recon-ng can automate and enhance information gathering.

Applications of Information Gathering

1. **Penetration Testing**: Security professionals use information gathering to identify and assess vulnerabilities in systems and networks as part of a penetration test.
2. **Threat Intelligence**: Gathering data on potential threats and adversaries helps organizations understand and prepare for possible attacks.
3. **Incident Response**: During a security incident, information gathering can help identify the scope of the breach and the potential impact.
4. **Network Defense**: Understanding the structure and weaknesses of your own network helps in strengthening defenses and mitigating risks.

Best Practices and Defensive Measures

1. **Limit Information Exposure**: Restrict the amount of sensitive information available publicly through privacy settings, internal policies, and careful management of online presence.
2. **Monitor and Analyze**: Continuously monitor network traffic and access logs to detect suspicious activities and respond to potential information gathering efforts.
3. **Regular Vulnerability Assessments**: Conduct regular scans and assessments to identify and address vulnerabilities before attackers can exploit them.

4. **Employee Training**: Educate employees about the importance of security and the risks of sharing too much information, both internally and externally.

Ethical Considerations

- **Legal Compliance**: Ensure all information gathering activities are conducted within the legal boundaries and with proper authorization. Unauthorized scanning or data collection can lead to legal consequences.
- **Privacy**: Respect privacy and handle collected data responsibly, especially when dealing with sensitive or personal information.

Information gathering is a foundational aspect of both offensive and defensive security strategies. By understanding how it works and implementing robust security practices, organizations can better protect themselves against potential threats.

FOOTPRINTING

Footprinting, also known as reconnaissance or information gathering, is the process of collecting as much information as possible about a target—such as a network, system, or organization—to identify potential vulnerabilities and plan attacks or defenses. It's a crucial step in both offensive and defensive security strategies. Here's an overview of footprinting, including methods, tools, and purposes:

Types of Footprinting

1. **Passive Footprinting**
    - **Objective**: To gather information without directly interacting with the target or causing any noticeable impact.
    - **Methods**:
        - **Search Engines**: Using search engines (e.g., Google, Bing) to find publicly available information. Techniques like Google Dorking can reveal hidden or sensitive information.
        - **Public Records**: Reviewing publicly accessible documents and databases, such as business filings, legal documents, and SEC reports.
        - **WHOIS Lookup**: Querying domain registration databases to obtain details about domain ownership, contact information, and registration dates.
        - **Social Media**: Analyzing social media profiles for information about the organization, employees, and their roles.
        - **DNS Queries**: Performing DNS lookups to gather details about domain names, IP addresses, and DNS records (e.g., MX, A, CNAME records).
2. **Active Footprinting**
    - **Objective**: To interact directly with the target to gather information, often using tools that can reveal live or hidden data.
    - **Methods**:
        - **Port Scanning**: Identifying open ports and services on a target system using tools like Nmap. This helps determine what services are running and their versions.
        - **Network Mapping**: Creating a map of the target network to understand the relationships between devices and services. Tools like Nmap and Zenmap are commonly used.
        - **OS Fingerprinting**: Identifying the operating system of a target machine to tailor subsequent attacks or defenses. Tools like Nmap and p0f can assist in this process.

- **Vulnerability Scanning**: Scanning systems for known vulnerabilities using tools such as Nessus or OpenVAS. This can help identify weaknesses in the target's defenses.

Footprinting Techniques and Tools

- **Search Engines**: Google, Bing
  - **Technique**: Querying for specific types of information, like file types or specific keywords, to locate sensitive data inadvertently exposed online.
  - **Tool**: Google Dorking for advanced search queries.
- **WHOIS Lookup**: WHOIS.net, DomainTools
  - **Technique**: Extracting domain registration details, including registrant contact information and domain expiration dates.
- **Social Media**: LinkedIn, Twitter, Facebook
  - **Technique**: Gathering information about employees, their roles, and their connections, which can be useful for social engineering.
- **DNS Tools**: nslookup, dig, MXToolbox
  - **Technique**: Querying DNS records to gather information about domain names, mail servers, and IP addresses.
- **Port Scanning Tools**: Nmap, Masscan
  - **Technique**: Scanning for open ports and services on a target system to identify potential entry points.
- **Network Mapping Tools**: Nmap, Zenmap
  - **Technique**: Mapping out the network structure to understand device relationships and network layout.
- **Vulnerability Scanners**: Nessus, OpenVAS
  - **Technique**: Scanning systems for known vulnerabilities and security weaknesses.

Purpose and Applications of Footprinting

1. **Penetration Testing**: Helps security professionals understand the target environment to identify and exploit vulnerabilities in a controlled manner.
2. **Threat Intelligence**: Assists in gathering information about potential threats and adversaries to improve security posture.
3. **Incident Response**: Provides insights into the scope and nature of security incidents by understanding the target environment.
4. **Network Defense**: Helps in identifying and addressing weaknesses within your own network before attackers can exploit them.

Defensive Measures

1. **Information Disclosure Controls**: Limit the amount of sensitive or critical information available publicly. Regularly review and update information exposure policies.
2. **Regular Security Assessments**: Conduct regular penetration tests and vulnerability assessments to identify and mitigate potential vulnerabilities.
3. **Employee Training**: Educate employees on the importance of protecting sensitive information and recognizing social engineering attempts.
4. **Network Monitoring**: Implement monitoring tools to detect and respond to suspicious activities and unauthorized scanning attempts.

By understanding and implementing effective footprinting techniques and defensive measures, organizations can better protect themselves against potential threats and vulnerabilities.

SCANNING IN INFORMATION GATHERING

Scanning in information gathering is a crucial step in the security assessment process. It involves actively probing a target system, network, or application to identify active components, open ports, services, and potential vulnerabilities. Scanning helps security professionals and attackers alike to gather detailed information that can be used to assess security weaknesses or plan an attack. Here's a detailed overview of scanning, including its types, methods, tools, and purposes:

Types of Scanning

1. **Network Scanning**
   - **Objective**: To identify active devices on a network, their IP addresses, and network services.
   - **Methods**:
     - **Ping Scanning**: Determines which hosts are active by sending ICMP Echo requests and checking for responses.
     - **Port Scanning**: Identifies open ports on a target system to determine which services are running.
     - **Service Scanning**: Detects the services running on open ports and their versions.
2. **Vulnerability Scanning**
   - **Objective**: To identify known vulnerabilities in systems, applications, and network services.
   - **Methods**:
     - **Automated Scanning**: Uses tools to scan systems for known vulnerabilities based on a database of exploits.
     - **Manual Scanning**: Involves manual testing and analysis to identify vulnerabilities that automated tools might miss.
3. **Application Scanning**
   - **Objective**: To identify vulnerabilities in web applications and other software.
   - **Methods**:
     - **Static Application Security Testing (SAST)**: Analyzes the application's source code or binaries for security flaws.
     - **Dynamic Application Security Testing (DAST)**: Tests the running application for vulnerabilities by sending various inputs and analyzing responses.
4. **OS Fingerprinting**
   - **Objective**: To identify the operating system of a target machine.
   - **Methods**:
     - **Passive Fingerprinting**: Observes traffic patterns and behaviors without actively probing the target.
     - **Active Fingerprinting**: Sends specific packets to the target and analyzes responses to determine the OS.

Common Scanning Tools

- **Nmap (Network Mapper)**
  - **Purpose**: Network discovery and security auditing.
  - **Features**: Port scanning, service detection, OS fingerprinting, network mapping.
  - **Usage**: nmap -sS [target] for SYN scan, nmap -sV [target] for service version detection.

- **Masscan**
  - **Purpose**: Fast port scanning.
  - **Features**: High-speed scanning of large networks.
  - **Usage**: masscan [target] -p[ports] for scanning specific ports.
- **Nessus**
  - **Purpose**: Vulnerability scanning.
  - **Features**: Automated vulnerability assessment with extensive plugin support.
  - **Usage**: Provides a web interface for configuring and running scans.
- **OpenVAS (Open Vulnerability Assessment System)**
  - **Purpose**: Comprehensive vulnerability scanning and management.
  - **Features**: Network and application vulnerability scanning, reporting.
  - **Usage**: Provides a web interface for configuring and running scans.
- **Burp Suite**
  - **Purpose**: Web application security testing.
  - **Features**: Intercepting proxy, scanner for web vulnerabilities.
  - **Usage**: Burp Suite provides tools for scanning and analyzing web applications.
- **Nikto**
  - **Purpose**: Web server scanning.
  - **Features**: Identifies vulnerabilities and misconfigurations in web servers.
  - **Usage**: nikto -h [target] for scanning web servers.

Purpose of Scanning

1. **Identifying Vulnerabilities**: To find weaknesses in systems, networks, or applications that could be exploited by attackers.
2. **Network Mapping**: To understand the structure and layout of the network, including the devices and services running.
3. **Security Assessment**: To assess the effectiveness of existing security measures and identify areas for improvement.
4. **Compliance Testing**: To ensure that systems and applications comply with security standards and regulations.

Best Practices for Scanning

1. **Authorization**: Ensure you have explicit permission to perform scanning on a network or system. Unauthorized scanning can be illegal and unethical.
2. **Scheduling**: Conduct scans during off-peak hours to minimize the impact on system performance and avoid disrupting operations.
3. **Regular Scanning**: Perform regular scans to detect new vulnerabilities and changes in the network or application environment.
4. **Data Analysis**: Carefully analyze scan results to prioritize and address vulnerabilities based on risk and impact.
5. **Use Multiple Tools**: Combine different scanning tools to get a comprehensive view of security issues and reduce the likelihood of missing critical vulnerabilities.

Defensive Measures

1. **Network Segmentation**: Implement network segmentation to limit the impact of a compromised system and contain potential threats.
2. **Patch Management**: Regularly update and patch systems to fix known vulnerabilities.

3. **Intrusion Detection Systems (IDS)**: Use IDS to monitor network traffic and detect suspicious activities or scanning attempts.
4. **Security Policies**: Develop and enforce policies for network and application security to mitigate risks and manage vulnerabilities effectively.

By understanding and effectively implementing scanning techniques and tools, organizations can enhance their security posture and better protect their systems and data from potential threats.

SOCIAL ENGINEERING

Social engineering in the context of information gathering is a technique used by attackers to collect sensitive or valuable information by manipulating individuals. This phase is crucial for attackers as it helps them gather the necessary details to plan and execute further attacks, such as phishing, pretexting, or even direct breaches. Here's a breakdown of how social engineering is used for information gathering and some common methods:

Methods of Social Engineering for Information Gathering

1. **Reconnaissance**: This is the preliminary phase where attackers collect as much information as possible about a target organization or individual. This can include public data, such as company structure, employee roles, and email addresses, obtained from social media profiles, company websites, or other public records.
2. **Phishing**: Attackers craft emails or messages that appear to be from a legitimate source to trick recipients into revealing sensitive information. These communications often include links or attachments designed to harvest credentials or other data.
3. **Pretexting**: The attacker creates a fake scenario to extract information. For example, they might pose as a new employee needing to update contact information or a representative from a trusted vendor asking for confirmation of certain details.
4. **Spear Phishing**: Unlike general phishing, spear phishing is targeted and personalized. Attackers use detailed information gathered from social media or other sources to craft convincing messages tailored to specific individuals within an organization.
5. **Baiting**: Involves offering something enticing, such as free software or a prize, to lure individuals into providing personal information or downloading malware. The bait is often tied to an angle that appeals to the target's interests or needs.
6. **Social Media Mining**: Attackers use information from social media platforms to gather data about individuals or organizations. They look for details such as job roles, personal interests, connections, or even login credentials if users have shared them.
7. **Phone Scams**: Attackers call targets pretending to be from legitimate organizations (e.g., tech support or customer service) and ask for personal or confidential information. This might involve asking for account numbers, passwords, or other sensitive data.
8. **Dumpster Diving**: This method involves searching through physical trash or discarded documents to find useful information. Attackers might look for old emails, internal reports, or other documents that could contain sensitive data.

Common Targets for Information Gathering

- **Employees**: Personal and professional information about employees, such as job titles, responsibilities, and contact information, can be used to craft more convincing attacks or gain unauthorized access to systems.
- **Company Infrastructure**: Details about the company's internal systems, network architecture, and security measures can be collected to identify vulnerabilities or to facilitate more targeted attacks.

- **Confidential Projects**: Information about current or upcoming projects, financial data, or business strategies can be valuable for competitive intelligence or corporate espionage.

Defensive Measures

1. **Employee Training**: Regularly train employees to recognize and handle social engineering attempts. This includes awareness of phishing tactics, handling unsolicited requests for information, and verifying the identity of those asking for sensitive data.
2. **Information Classification**: Implement data classification policies to protect sensitive information and limit access to it based on necessity.
3. **Monitoring and Reporting**: Encourage employees to report any suspicious requests or incidents. Implement monitoring systems to detect unusual or unauthorized access patterns.
4. **Privacy Settings**: Educate employees on the importance of maintaining privacy on social media and other online platforms. Ensure that company-related information is not overly accessible or detailed in public domains.
5. **Verification Procedures**: Establish and enforce strict procedures for verifying requests for sensitive information, whether they come via email, phone, or in person.

By understanding and mitigating the risks associated with social engineering in information gathering, organizations and individuals can better protect themselves from security threats and attacks.

NMAP (NETWORK MAPPER)

Nmap (Network Mapper) is a powerful open-source tool used for network discovery and security auditing. It's widely employed by network administrators, security professionals, and even hackers to gather information about networked devices and services. Here's a brief overview of its key features and functionalities:

Key Features of Nmap:

1. **Network Discovery**: Nmap can identify devices on a network, their IP addresses, and the services they are running. This helps in understanding the network layout and detecting unauthorized devices.
2. **Port Scanning**: It can scan for open ports on a host. Knowing which ports are open can help in identifying which services are running and could be vulnerable to attacks.
3. **Service Version Detection**: Nmap can determine the version of the services running on open ports, which is useful for identifying vulnerabilities specific to certain software versions.
4. **OS Detection**: By analyzing various characteristics of the network responses, Nmap can often determine the operating system and its version running on a target host.
5. **Scriptable Interactions**: With the Nmap Scripting Engine (NSE), users can write scripts to automate a variety of network tasks and security assessments, including vulnerability detection and brute-force attacks.
6. **Host Discovery**: It can identify hosts that are online by sending various types of packets and analyzing responses.
7. **Network Mapping**: Nmap can create a map of a network, identifying the devices and their interconnections.

Common Nmap Commands:

- **Basic Scan**:

  Php code

nmap <target>

This performs a default scan on the specified target, usually identifying open ports and services.

- **Scan Multiple IPs or Subnets**:

  Php code
  nmap <target1> <target2> ... <targetN>
  nmap 192.168.1.0/24

  Scans multiple targets or an entire subnet.

- **Port Scan with Specific Ports**:

  Css code
  nmap -p 22,80,443 <target>

  Scans only the specified ports.

- **Service Version Detection**:

  Php code
  nmap -sV <target>

  Detects versions of the services running on open ports.

- **Operating System Detection**:

  Mathematica code
  nmap -O <target>

  Attempts to determine the operating system of the target.

- **Aggressive Scan**:

  Css code
  nmap -A <target>

  Performs a comprehensive scan including OS detection, service version detection, and script scanning.

- **Using Nmap Scripting Engine**:

  Php code
  nmap --script <script_name> <target>

  Runs specific scripts for various tasks like vulnerability detection.

Use Cases:

1. **Network Security Auditing**: Identifying vulnerabilities and potential security issues in your own network.
2. **Penetration Testing**: Assisting ethical hackers in assessing network security.
3. **Network Inventory**: Keeping track of devices and services running on a network.
4. **Compliance**: Ensuring systems meet security standards and compliance requirements.

While Nmap is a powerful tool, it should be used responsibly. Scanning networks or systems you do not own or have explicit permission to test can be illegal and unethical. Always ensure you have authorization before conducting any scans.

Nmap's flexibility and extensive feature set make it an invaluable tool in both network management and security analysis.

ZENMAP

Zenmap is the official graphical user interface (GUI) for Nmap. It provides a more user-friendly way to interact with Nmap's powerful scanning capabilities, making it accessible to users who may not be as comfortable with command-line tools. Here's a breakdown of what Zenmap offers:

Key Features of Zenmap:

1. **Graphical Interface**: Zenmap provides a graphical front-end to Nmap, allowing users to perform network scans, view results, and manage profiles without needing to use the command line.
2. **Scan Profiles**: It allows users to create and save scan profiles, which are predefined sets of scan options. This makes it easy to repeat common scan types with just a few clicks.
3. **Scan Results Visualization**: Zenmap offers various ways to view and interpret scan results, including graphical maps and detailed reports, which help in understanding the network topology and the security status of devices.
4. **Command Editor**: Users can see and edit the underlying Nmap command for each scan, providing transparency and the ability to tweak options directly.
5. **Network Topology Visualization**: Zenmap can generate visual representations of network topology based on scan results, which can be useful for network mapping and visualization.
6. **Comparison of Results**: It allows users to compare scan results over time or between different scans, helping to track changes and detect anomalies.
7. **Security Auditing and Reporting**: Zenmap can generate reports based on the scan results, which can be useful for security audits and documentation.

Using Zenmap:

1. **Starting a Scan**:
    - Launch Zenmap.
    - Enter the target IP address or hostname in the "Target" field.
    - Select or customize a scan profile from the "Profile" dropdown menu or by modifying the scan options directly.
    - Click the "Scan" button to start the scan.
2. **Viewing Results**:
    - After the scan completes, results are displayed in various tabs, including "Nmap Output," "Ports/Hosts," "Topology," and "Host Details."
    - You can explore open ports, services, and other details in these tabs.

3. **Saving and Managing Profiles**:
    - Create and manage scan profiles using the “Profile” menu.
    - Save frequently used scan configurations for easy access in the future.
4. **Generating Reports**:
    - Use the “Save” or “Export” options to generate reports based on the scan results. Zenmap supports various formats, including XML and plain text.

Benefits of Using Zenmap:

- **Ease of Use**: The graphical interface simplifies interaction with Nmap, especially for users less familiar with command-line operations.
- **Visualization**: Provides intuitive visualizations of network data and scan results, making it easier to understand complex information.
- **Efficiency**: Allows for quick execution and management of repeated scans with saved profiles and easy access to results.

Limitations:

- **Performance**: Zenmap can be less performant compared to using Nmap directly from the command line, especially for very large scans.
- **Complexity**: While it simplifies many tasks, complex scanning options may still require familiarity with Nmap’s command-line syntax.

Overall, Zenmap is a valuable tool for those who prefer a GUI over the command line and need a robust way to perform and manage network scans using Nmap.

Example of using Zenmap

Here's a simple example of how to use Zenmap to perform a basic network scan:

1. **Open Zenmap**: Launch the Zenmap application on your computer.
2. **Enter Target**: In the “Target” field, enter the IP address or hostname of the target you want to scan. For example, use 192.168.1.1 to scan a specific IP address.
3. **Select Profile**:
    - Choose a scan profile from the “Profile” dropdown menu. For a basic scan, select "Intense scan" to perform a comprehensive scan, or use "Quick scan" for a faster overview.
    - You can also create a custom profile if needed by modifying the scan options directly.
4. **Start Scan**: Click the “Scan” button to initiate the scan. Zenmap will run Nmap with the selected profile's settings.
5. **View Results**:
    - **Nmap Output**: Check the “Nmap Output” tab for detailed command-line output from the scan.
    - **Ports/Hosts**: The “Ports/Hosts” tab displays the open ports and associated services on the target.
    - **Topology**: The “Topology” tab shows a visual representation of the network topology, if applicable.
    - **Host Details**: The “Host Details” tab provides detailed information about the target host.
6. **Save Results**: Use the “Save” or “Export” options to save the scan results for later review or reporting.

This process helps you quickly gather and analyze information about your network or any other target using Zenmap's graphical interface.

PORT SCANNERS

Port scanners are tools used to identify open ports and services on a networked device. They play a crucial role in network security, administration, and troubleshooting by providing insight into which ports are accessible and which services are running on them. Here's an overview of port scanners, including how they work, their types, and their use cases:

How Port Scanners Work

1. **Sending Requests**: A port scanner sends packets to a range of ports on a target device. Each packet is crafted to provoke a response from the target, indicating whether a port is open or closed.
2. **Analyzing Responses**: The scanner analyzes responses from the target to determine the status of each port. Common responses include SYN-ACK (indicating an open port), RST (indicating a closed port), or no response at all.
3. **Reporting Results**: After scanning, the scanner generates a report listing open ports, the services running on those ports, and other relevant information.

Types of Port Scanners

1. **TCP Connect Scanner**:
    - **How It Works**: Attempts to establish a full TCP connection (three-way handshake) with each port.
    - **Pros**: Reliable and straightforward.
    - **Cons**: More easily detected and can be slower due to the full handshake process.
2. **SYN Scanner (Half-Open Scan)**:
    - **How It Works**: Sends SYN packets to target ports and waits for responses (SYN-ACK or RST). It doesn't complete the handshake, which can make it less detectable.
    - **Pros**: Faster and less likely to be logged.
    - **Cons**: Less accurate on some networks and might require higher privileges.
3. **UDP Scanner**:
    - **How It Works**: Sends UDP packets to target ports and listens for responses. Since UDP is connectionless, it's harder to determine the status of a port.
    - **Pros**: Can detect open UDP ports.
    - **Cons**: Can be slow and less reliable due to the nature of UDP.
4. **Stealth Scanners**:
    - **How They Work**: Use techniques to evade detection, such as sending fragmented packets or randomizing packet timings.
    - **Pros**: Harder to detect by intrusion detection systems (IDS).
    - **Cons**: Can be complex and might still be detected.
5. **Comprehensive Scanners**:
    - **How They Work**: Combine various scanning techniques to gather detailed information about ports and services.
    - **Pros**: Provide thorough and detailed results.
    - **Cons**: Can be resource-intensive and may take longer to run.

Popular Port Scanning Tools

1. **Nmap**:
    - **Description**: A widely used and versatile port scanner that supports multiple scanning techniques and includes features for service version detection, OS fingerprinting, and more.
    - **Command Example**: nmap -p 22,80,443 192.168.1.1
2. **Zenmap**:
    - **Description**: The GUI frontend for Nmap, offering an easier interface for running and managing scans.
    - **Command Example**: Uses profiles and graphical interface for scans, such as performing an "Intense scan" on a target.
3. **Netcat (nc)**:
    - **Description**: A network utility that can be used for port scanning in addition to its other functions like data transfer.
    - **Command Example**: nc -zv 192.168.1.1 1-65535 (scans all ports on the target)
4. **Angry IP Scanner**:
    - **Description**: A lightweight, cross-platform tool for scanning IP addresses and ports.
    - **Command Example**: Uses a GUI to input ranges and perform scans.

Use Cases for Port Scanners

1. **Network Security Assessment**: Identifying open ports and services to find potential vulnerabilities.
2. **Network Inventory**: Mapping out devices and their services on a network.
3. **Troubleshooting**: Diagnosing network issues by identifying misconfigured or non-responsive ports.
4. **Compliance**: Ensuring that only authorized ports are open and that security policies are being followed.

Limitations

- **Authorization**: Always ensure you have permission before scanning networks or systems that you do not own.
- **Legal Compliance**: Unauthorized scanning can be illegal and could result in serious consequences. Conduct scans responsibly and ethically.

Port scanners are powerful tools for understanding network configurations and security postures. They help network administrators manage their infrastructure and security professionals assess vulnerabilities and compliance.

NETWORK SCANNERS

Network scanners are tools used to discover devices, map networks, and assess the health and security of networked systems. They provide valuable insights into network topology, connected devices, and potential vulnerabilities. Here's an overview of network scanners, their types, and use cases:

Types of Network Scanners

1. **Network Discovery Scanners**:

- **Function**: Identify devices on a network, their IP addresses, and their associated information.
- **Examples**:
  - **Nmap**: Offers extensive discovery features including host discovery and network mapping.
  - **Angry IP Scanner**: Simple tool for discovering devices on a network.

2. **Port Scanners**:
   - **Function**: Identify open ports and services running on network devices.
   - **Examples**:
     - **Nmap**: Provides detailed port scanning capabilities and service detection.
     - **Zenmap**: GUI for Nmap that simplifies port scanning and results analysis.
3. **Vulnerability Scanners**:
   - **Function**: Identify security vulnerabilities in networked devices and services.
   - **Examples**:
     - **Nessus**: Comprehensive vulnerability scanner that provides detailed reports and remediation advice.
     - **OpenVAS**: Open-source vulnerability scanner with extensive testing capabilities.
     -
4. **Network Performance Monitors**:
   - **Function**: Measure and analyze network performance metrics like bandwidth, latency, and packet loss.
   - **Examples**:
     - **Nagios**: Monitors network performance and provides alerts for issues.
     - **PRTG Network Monitor**: Offers detailed performance monitoring and visualization.
5. **Network Mapping Tools**:
   - **Function**: Create visual maps of network devices and their interconnections.
   - **Examples**:
     - **SolarWinds Network Topology Mapper**: Provides automatic network mapping and visualization.
     - **NetBrain**: Offers dynamic network mapping and documentation.

Use Cases for Network Scanners

1. **Network Inventory**: Discover all devices connected to the network, including routers, switches, servers, and workstations.
2. **Network Security**: Identify potential vulnerabilities and security risks by scanning for open ports, outdated services, and misconfigurations.
3. **Troubleshooting**: Diagnose network issues by analyzing device connectivity, performance metrics, and service status.
4. **Compliance**: Ensure that network configurations meet regulatory and security standards by regularly scanning and assessing network assets.

Popular Network Scanning Tools

1. **Nmap**: Versatile tool for network discovery, port scanning, and service detection. It supports various scanning techniques and scripting.
2. **Wireshark**: A network protocol analyzer that captures and inspects network traffic in real-time.
3. **Advanced IP Scanner**: Provides fast network discovery and IP address management.
4. **Fping**: A tool for pinging multiple hosts to check their status and response times.

Limitations

- **Authorization**: Ensure you have permission before scanning networks or devices you do not own.
- **Legal Compliance**: Unauthorized network scanning can be illegal and may lead to legal consequences. Always perform scans within the bounds of law and ethics.

Network scanners are essential for maintaining network security and performance. They help administrators and security professionals gain insights into network structures, detect potential issues, and ensure that systems operate efficiently and securely.

UNIT – II EXPLANATION OF MALWARE, TYPES OF MALWARE:

Virus, Worms, Trojans, Rootkits, Robots, Adware's, Spywares, Ransom wares, Zombies etc.,, Malware Analysis. Open Source/ Free/ Trial Tools: Antivirus Protection, Anti Spywares, System tuning tools, Anti Phishing.

MALWARE - INTRODUCTION

Malware is malicious software and refers to any software that is designed to cause harm to computer systems, networks, or users. Malware can take many forms. Individuals and organizations need to be aware of the different types of malware and take steps to protect their systems, such as using antivirus software, keeping software and systems up-to-date, and being cautious when opening email attachments or downloading software from the internet.

What is Malware?

Malware is software that gets into the system without user consent to steal the user's private and confidential data, including bank details and passwords. They also generate annoying pop-up ads and change system settings. Malware includes computer viruses, worms, Trojan horses, ransomware, spyware, and other malicious programs. Individuals and organizations need to be aware of the different types of malware and take steps to protect their systems, such as using antivirus software, keeping software and systems up-to-date, and being cautious when opening email attachments or downloading software from the internet.

What Does Malware Do?

Malware is designed to harm and exploit your computer or network. It can steal sensitive information like passwords and credit card numbers, disrupt your system's operations, and even allow attackers to gain unauthorized access to your device. Some types of malware, such as ransomware, encrypt your files and demand payment to unlock them, while spyware monitors your activities and sends the information back to the attacker. Additionally, malware can spread to other devices on the same network, making it a significant threat. Protecting your devices with up-to-date antivirus software and being cautious about your open links and attachments can help mitigate these risks.

Why Do Cybercriminals Use Malware?

Cybercriminals use malware, including all forms of malicious software including viruses, for various purposes.

- Using deception to induce a victim to provide personal information for identity theft
- Theft of customer credit card information or other financial information
- Taking over several computers and using them to launch denial-of-service attacks against other networks
- Using infected computers to mine for cryptocurrencies like bitcoin.

TYPES OF MALWARE

Viruses – A Virus is a malicious executable code attached to another executable file. The virus spreads when an infected file is passed from system to system. Viruses can be harmless or they can modify or delete data. Opening a file can trigger a virus. Once a program virus is active, it will infect other programs on the computer.

Worms – Worms replicate themselves on the system, attaching themselves to different files and looking for pathways between computers, such as computer network that shares common file storage areas. Worms usually slow down networks. A virus needs a host program to run but worms can run by themselves. After a worm affects a host, it is able to spread very quickly over the network.

Trojan horse – A Trojan horse is malware that carries out malicious operations under the appearance of a desired operation such as playing an online game. A Trojan horse varies from a virus because the Trojan binds itself to non-executable files, such as image files, and audio files.

Ransomware – Ransomware grasps a computer system or the data it contains until the victim makes a payment. Ransomware encrypts data in the computer with a key that is unknown to the user. The user has to pay a ransom (price) to the criminals to retrieve data. Once the amount is paid the victim can resume using his/her system.

Adware – It displays unwanted ads and pop-ups on the computer. It comes along with software downloads and packages. It generates revenue for the software distributer by displaying ads.

Spyware – Its purpose is to steal private information from a computer system for a third party. Spyware collects information and sends it to the hacker.

Logic Bombs – A logic bomb is a malicious program that uses a trigger to activate the malicious code. The logic bomb remains non-functioning until that trigger event happens. Once triggered, a logic bomb implements a malicious code that causes harm to a computer. Cybersecurity specialists recently discovered logic bombs that attack and destroy the hardware components in a workstation or server including the cooling fans, hard drives, and power supplies. The logic bomb overdrives these devices until they overheat or fail.

Rootkits – A rootkit modifies the OS to make a backdoor. Attackers then use the backdoor to access the computer distantly. Most rootkits take advantage of software vulnerabilities to modify system files.

Backdoors – A backdoor bypasses the usual authentication used to access a system. The purpose of the backdoor is to grant cyber criminals future access to the system even if the organization fixes the original vulnerability used to attack the system.

Keyloggers – Keylogger records everything the user types on his/her computer system to obtain passwords and other sensitive information and send them to the source of the keylogging program.

How To Know If Our Devices Are Infected With Malware?

Performing poorly on the computer by execution.

When your web browser directs you to a website you didn’t intend to visit, this is known as a browser redirect.

Warnings about infections are frequently accompanied by offers to buy a product to treat them.

Having trouble starting or shutting down your computer. Persistent pop-up ads.

How To Protect From Malware?

- Update your operating system and software. Install updates as soon as they become available because cybercriminals search for vulnerabilities in out-of-date or outdated software.
- Never click on a popup's link. Simply click the "X" in the message's upper corner to close it and leave the page that generated it.
- Don't install too many apps on your devices. Install only the apps you believe you will regularly use and need.
- Be cautious when using the internet.
- Do not click on unidentified links. If a link seems suspicious, avoid clicking it whether it comes from an email, social networking site, or text message.
- Choose the websites you visit wisely. Use a safe search plug-in and try to stick to well-known and reputable websites to avoid any that might be malicious without your knowledge.
- Emails requesting personal information should be avoided. Do not click a link in an email that appears to be from your bank and asks you to do so in order to access your account or reset your password. Log in immediately at your online banking website.

How To Remove Malware?

A large number of security software programs are made to both find and stop malware as well as to eliminate it from infected systems. An antimalware tool that handles malware detection and removal is Malwarebytes. Malware can be eliminated from Windows, macOS, Android, and iOS operating systems. A user's registry files, currently running programs, hard drives, and individual files can all be scanned by Malwarebytes. Malware can then be quarantined and removed if it is found. Users cannot, however, set automatic scanning schedules like they can with some other tools.

Tools Used to Remove Malware

- Malwarebytes
- SUPERAntiSpyware
- Malicious Software Removal Tool (MSRT)
- Bitdefender Antivirus Free Edition
- Adaware Antivirus Free
- Avast Free Mac Security

Advantages of Detecting and Removing Malware

Improved Security: By detecting and removing malware, individuals, and organizations can improve the security of their systems and reduce the risk of future infections.

Prevent Data Loss: Malware can cause data loss, and by removing it, individuals and organizations can protect their important files and information.

Protect Reputation: Malware can cause harm to a company's reputation, and by detecting and removing it, individuals and organizations can protect their image and brand.

Increased Productivity: Malware can slow down systems and make them less efficient, and by removing it, individuals and organizations can increase the productivity of their systems and employees.

Disadvantages of Detecting and Removing Malware

Time-Consuming: The process of detecting and removing malware can be time-consuming and require specialized tools and expertise.

Cost: Antivirus software and other tools required to detect and remove malware can be expensive for individuals and organizations.

False Positives: Malware detection and removal tools can sometimes result in false positives, causing unnecessary alarm and inconvenience.

Difficulty: Malware is constantly evolving, and the process of detecting and removing it can be challenging and require specialized knowledge and expertise.

Risk of Data Loss: Some malware removal tools can cause unintended harm, resulting in data loss or system instability.

MALWARE ANALYSIS

Malware analysis is the process of examining malicious software (malware) to understand its behavior, capabilities, and impact. This analysis can be crucial for developing strategies to protect systems, respond to incidents, and improve security measures. There are different approaches and techniques used in malware analysis, including:

1. **Static Analysis:**
    - **Code Review:** Examining the code of the malware without executing it. This involves disassembling or decompiling the malware to understand its structure and functionality.
    - **Signature Analysis:** Identifying known patterns or signatures in the malware code that match those in a database of known malware.
2. **Dynamic Analysis:**
    - **Behavioral Analysis:** Running the malware in a controlled environment (sandbox) to observe its behavior, such as file modifications, network activity, or changes to system settings.
    - **System Call Analysis:** Monitoring system calls made by the malware to understand its interactions with the operating system.
3. **Reverse Engineering:**
    - **Disassembly:** Converting the machine code of the malware into assembly language to understand its functionality.
    - **Decompilation:** Translating the binary code into a higher-level language to gain insight into the program's logic.
4. **Network Analysis:**
    - **Traffic Monitoring:** Analyzing network traffic generated by the malware to identify command-and-control servers, data exfiltration, or other malicious activities.
    - **Packet Analysis:** Inspecting network packets to understand how the malware communicates and what data it exchanges.
5. **Forensic Analysis:**
    - **File Analysis:** Investigating files associated with the malware, such as dropped files or registry entries.
    - **Memory Analysis:** Examining the memory of a system for artifacts left by the malware, such as injected code or data structures.
6. **Behavioral Analysis:**
    - **Execution Tracing:** Observing and documenting the sequence of actions taken by the malware during execution.

The ultimate goal of malware analysis is to develop effective detection methods, mitigate the impact of the malware, and enhance overall cybersecurity defenses.

ANTIVIRUS PROTECTION

Antivirus protection is crucial for safeguarding your computer or device from malicious software, commonly known as malware. Here's a breakdown of how it works and why it's important:

How Antivirus Protection Works

1. **Scanning and Detection**:
    - **Signature-Based Detection**: Antivirus programs compare files and programs on your device against a database of known malware signatures (unique patterns or code snippets).
    - **Heuristic-Based Detection**: This method involves analyzing the behavior and characteristics of programs to identify potentially harmful activities that are not yet in the signature database.
    - **Behavioral-Based Detection**: Monitors the behavior of programs in real-time. If a program starts acting in a way that is typically associated with malware, the antivirus can take action.
2. **Real-Time Protection**:
    - Continuously monitors your system for signs of malware, blocking and removing threats as they are detected.
3. **Regular Updates**:
    - Antivirus software is regularly updated to include new virus definitions and improve detection capabilities against the latest threats.
4. **Quarantine**:
    - Suspicious files are placed in a quarantine zone, isolating them from the rest of your system while you decide whether to delete them or restore them.
5. **Removal Tools**:
    - Once a threat is detected, antivirus software can often remove or clean the infected files to restore your system to a secure state.

Why Antivirus Protection is Important

1. **Prevents Data Loss**:
    - Malware can corrupt or delete important files. Antivirus software helps prevent this by catching threats before they cause harm.
2. **Protects Privacy**:
    - Some malware is designed to steal personal information, such as login credentials or financial details. Effective antivirus protection can guard against these types of attacks.
3. **Maintains System Performance**:
    - Malware can slow down your computer by using system resources. By preventing infections, antivirus software helps keep your system running smoothly.
4. **Safeguards Against Various Threats**:
    - Antivirus programs protect against a wide range of threats, including viruses, worms, trojans, ransomware, spyware, and adware.
5. **Reduces Risk of Network Infection**:
    - If your device is infected, it can potentially spread malware to other devices on the same network. Antivirus protection helps mitigate this risk.

Choosing Antivirus Software

When selecting antivirus software, consider the following factors:

- **Detection Rates**: Look for software with high detection and low false-positive rates.
- **System Impact**: Choose a solution that doesn't significantly slow down your system.
- **Features**: Some antivirus programs offer additional features such as firewall protection, VPN services, and secure browsing tools.
- **Cost**: There are both free and paid options available. Paid solutions often come with more advanced features and better support.

Having robust antivirus protection is a key component of a comprehensive cybersecurity strategy, helping to ensure that your digital life remains secure.

ANTI - SPYWARES

Anti-spyware software is designed to protect your computer or device from spyware and other types of malicious software that can collect personal information without your consent. Here's a detailed look at what anti-spyware does and why it's important:

What is Spyware?

Spyware is a type of malware that secretly monitors and collects information about a user's activities and personal data without their knowledge. This can include:

- **Keystrokes**: Logging what you type, such as passwords or credit card numbers.
- **Screen Captures**: Taking screenshots of your activity.
- **Personal Information**: Gathering sensitive data like addresses, phone numbers, and financial details.
- **Browser History**: Tracking websites you visit and your online behavior.

How Anti-Spyware Works

1. **Detection and Scanning**:
    - **Signature-Based Detection**: Identifies known spyware by comparing files and behaviors to a database of known spyware signatures.
    - **Heuristic-Based Detection**: Analyzes the behavior of programs and their code to identify new or unknown spyware that exhibits suspicious behavior.
    - **Behavioral-Based Detection**: Monitors system activities in real-time to spot behaviors characteristic of spyware, such as unauthorized data access.
2. **Real-Time Protection**:
    - Continuously monitors your system for any signs of spyware or suspicious activities, blocking and alerting you if any threats are detected.
3. **Removal and Quarantine**:
    - When spyware is detected, anti-spyware software can remove or quarantine it to prevent it from causing further harm. Quarantine isolates the spyware so it can't affect your system, giving you the option to delete it later.
4. **Regular Updates**:
    - Anti-spyware programs regularly update their databases to include new spyware definitions and improve their ability to detect the latest threats.
5. **Privacy Protection**:
    - Many anti-spyware tools offer features to help protect your privacy, such as blocking tracking cookies and preventing unauthorized access to personal information.

Why Anti-Spyware is Important

1. **Protects Personal Information**:
   - Anti-spyware software helps prevent the theft of sensitive personal data, such as login credentials, financial information, and private communications.
2. **Enhances Privacy**:
   - By detecting and removing spyware, these tools help ensure that your online activities and personal data remain private.
3. **Prevents Identity Theft**:
   - Spyware that collects personal and financial information can be used for identity theft. Anti-spyware helps mitigate this risk by keeping such threats at bay.
4. **Improves System Performance**:
   - Spyware can slow down your computer by consuming system resources and affecting performance. Anti-spyware software helps keep your system running smoothly by removing these threats.
5. **Reduces Risk of Other Malware**:
   - Spyware often operates alongside other types of malware. By removing spyware, you can reduce the risk of encountering additional threats.

Choosing Anti-Spyware Software

When selecting anti-spyware software, consider the following:

- **Detection Rates**: Look for software with a high detection rate for spyware and minimal false positives.
- **System Impact**: Choose a solution that runs efficiently without significantly impacting your system's performance.
- **Features**: Some anti-spyware programs offer additional features like anti-virus protection, firewall capabilities, and secure browsing tools.
- **Cost**: There are both free and paid anti-spyware options. Paid versions often provide more comprehensive protection and customer support.

SYSTEM TUNING TOOLS

System tuning tools are software utilities designed to optimize the performance and efficiency of your computer or device. They can help you manage system resources, improve speed, enhance stability, and even address security concerns. Here's a detailed look at the different types of system tuning tools and some popular options:

Types of System Tuning Tools

1. **Performance Optimization Tools**
   - **CPU and Memory Management**: Tools that help monitor and optimize CPU usage and memory allocation.
   - **Disk Optimization**: Utilities for defragmenting hard drives (for HDDs) and managing disk space.
2. **Startup Management Tools**
   - **Startup Program Managers**: Tools that allow you to control which applications and processes run at startup, helping to speed up boot times.
3. **System Cleanup Tools**
   - **Junk File Cleaners**: Remove temporary files, system caches, and other unnecessary files to free up disk space.
   - **Registry Cleaners**: Fix or remove invalid or obsolete registry entries (primarily for Windows).

4. **System Monitoring Tools**
    - **Resource Monitors**: Track CPU, memory, disk, and network usage in real-time to identify performance bottlenecks.
    - **Diagnostic Tools**: Tools that provide detailed reports and diagnostics on system health and performance.
5. **Power Management Tools**
    - **Power Configuration Managers**: Optimize power settings for better performance or energy savings.
6. **Security and Privacy Tools**
    - **Privacy Cleaners**: Manage and delete browsing history, cookies, and other privacy-related data.
    - **Antivirus and Antimalware Tools**: Ensure that your system is protected from malicious software.

Popular System Tuning Tools

Windows

1. **CCleaner**
    - **Features**: Cleans junk files, manages startup programs, and includes a basic registry cleaner.
    - **Usage**: Improves system performance and frees up disk space.
2. **Advanced SystemCare**
    - **Features**: Comprehensive tool with performance optimization, privacy protection, and malware removal.
    - **Usage**: Provides an all-in-one solution for system tuning and security.
3. **Glary Utilities**
    - **Features**: Includes disk cleanup, registry repair, memory optimization, and more.
    - **Usage**: A versatile tool for maintaining and optimizing system performance.
4. **IObit Driver Booster**
    - **Features**: Scans for outdated drivers and updates them automatically.
    - **Usage**: Helps ensure that your system's drivers are up-to-date for better stability and performance.

Cross-Platform

1. **System Mechanic**
    - **Features**: Offers performance optimization, privacy protection, and security tools.
    - **Usage**: Available for both Windows and macOS, it provides a comprehensive approach to system maintenance.
2. **TuneUp Utilities**
    - **Features**: Provides disk cleanup, registry cleaning, and performance optimization tools.
    - **Usage**: While less common now, it was a popular tool for system maintenance and tuning.
3. **Glary Utilities (also available for macOS)**
    - **Features**: Offers a wide range of utilities for cleanup, optimization, and repair.
    - **Usage**: A robust option for system tuning on both Windows and macOS platforms.

By leveraging system tuning tools effectively, you can enhance your computer’s performance, maintain stability, and ensure a smoother user experience.

ANTI-PHISHING

Anti-phishing refers to the practices, tools, and technologies designed to protect users from phishing attacks. Phishing is a type of cyber attack where attackers impersonate legitimate organizations or individuals to deceive people into divulging sensitive information, such as usernames, passwords, credit card numbers, or other personal data. Here's a comprehensive overview of anti-phishing strategies, tools, and best practices:

How Phishing Attacks Work

1. **Deceptive Emails or Messages**:
   - **Spoofed Email Addresses**: Attackers use email addresses that closely resemble legitimate ones.
   - **Urgent or Threatening Messages**: Emails may create a sense of urgency or fear, urging recipients to act quickly.
   - **Fake Links**: Messages often include links that lead to fraudulent websites designed to capture user credentials.
2. **Malicious Websites**:
   - **Clone Websites**: Attackers create fake versions of legitimate websites to trick users into entering their login credentials or other sensitive information.
   - **SSL Certificates**: Some phishing sites use HTTPS to appear secure, even though they are not.
3. **Social Engineering**:
   - **Impersonation**: Attackers may pose as trusted figures or companies to gain trust and obtain sensitive information.
   - **Personalization**: Phishing attempts may use information gathered from social media or previous interactions to make attacks more convincing.

Anti-Phishing Tools and Technologies

1. **Email Filtering**:
   - **Spam Filters**: Advanced email filters can identify and block phishing emails based on content, sender reputation, and known phishing patterns.
   - **Email Security Gateways**: Specialized solutions that provide comprehensive protection against phishing and other email-borne threats.
2. **Web Filtering**:
   - **URL Filtering**: Blocks access to known phishing websites by comparing URLs against databases of known threats.
   - **Domain Reputation Services**: Evaluate the trustworthiness of domains to identify and block access to malicious sites.
3. **Browser Extensions and Add-ons**:
   - **Anti-Phishing Extensions**: Tools like the Netcraft Anti-Phishing Toolbar or similar browser extensions can provide warnings about potentially dangerous websites.
   - **Password Managers**: Many password managers have built-in anti-phishing features that help identify and prevent entry of credentials on fraudulent sites.
4. **Multi-Factor Authentication (MFA)**:
   - **Enhanced Security**: MFA adds an extra layer of security by requiring additional verification (e.g., a code sent to your phone) in addition to your password.
5. **Security Awareness Training**:
   - **User Education**: Regular training programs to educate users about recognizing phishing attempts and safe online practices.
   - **Simulated Phishing Campaigns**: Conducting simulated phishing attacks to test and improve employee awareness and response.

Best Practices for Anti-Phishing

1. **Verify Suspicious Communications**:
    - **Check Email Addresses**: Verify the sender's email address and look for subtle discrepancies.
    - **Do Not Click Links**: Avoid clicking on links in unsolicited emails or messages. Instead, visit websites directly by typing their URL into the browser.
2. **Use Strong, Unique Passwords**:
    - **Password Management**: Utilize password managers to create and store strong, unique passwords for each of your accounts.
3. **Enable MFA**:
    - **Additional Layer of Protection**: Implement MFA on accounts that support it to provide extra security in case your credentials are compromised.
4. **Keep Software Updated**:
    - **Patch Management**: Ensure that your operating system, browsers, and security software are up-to-date to protect against known vulnerabilities.
5. **Report Phishing Attempts**:
    - **Notify Relevant Parties**: Report phishing attempts to your IT department, email provider, or other relevant authorities to help prevent further attacks.
6. **Educate Yourself and Others**:
    - **Awareness Training**: Stay informed about common phishing tactics and share this knowledge with colleagues, friends, and family.

Popular Anti-Phishing Tools

- **For Email**: **Proofpoint and Barracuda Email Security Gateway**: Offers comprehensive email filtering and anti-phishing protection.
- **For Browsers**: **Netcraft Extension**and **Malwarebytes Browser Guard**
- **For Security Awareness**: **KnowBe4** and **PhishMe**: Provides training and simulation tools to help organizations recognize and respond to phishing attacks.

By implementing robust anti-phishing measures and staying vigilant, you can significantly reduce the risk of falling victim to phishing attacks and protect your sensitive information.

UNIT – III SECURITY IN CONVENTIONAL OPERATING SYSTEMS:

Memory, time, file, object protection requirements and techniques - Identification and authentication - Trusted operating systems.

MEMORY

Memory security in operating systems is a crucial aspect of ensuring the overall safety and integrity of computing environments. Here's an overview of key concepts and practices related to memory security:

1. Memory Management

Virtual Memory:

- Virtual memory allows an OS to use hardware and software to compensate for physical memory shortages. This helps in managing memory more efficiently and securely by isolating processes and preventing them from directly accessing each other's memory space.

Paging and Segmentation:

- Paging breaks memory into fixed-size pages, while segmentation divides memory into variable-sized segments. Both methods help in managing memory allocation and protection.

2. Memory Protection Mechanisms

Address Space Layout Randomization (ASLR):

- ASLR randomizes the locations of key data areas in a process's address space, such as the base addresses of executables, heap, and stack. This makes it harder for attackers to predict the location of specific memory regions and exploit vulnerabilities.

Data Execution Prevention (DEP) / Execute Disable (XD):

- DEP/XD prevents code from executing in certain regions of memory that are not designated for execution, such as the stack or heap. This helps mitigate attacks that attempt to execute code in non-executable memory regions.

Stack Canaries:

- Stack canaries are special values placed on the stack that help detect buffer overflows. If an overflow occurs, the canary value changes, signaling that a buffer overflow attack might be underway.

3. Access Control and Isolation

Process Isolation:

- Modern operating systems use process isolation to ensure that each process runs in its own memory space. This prevents processes from interfering with each other and protects sensitive data.

User and Kernel Mode:

- Operating systems typically operate in user mode and kernel mode. User mode is restricted and isolated from kernel mode, which has full access to the hardware. This separation helps in protecting the OS from potentially malicious user-space code.

Memory Protection Units (MPUs) and Memory Management Units (MMUs):

- MPUs and MMUs are hardware components that enforce memory protection policies and manage address translation. They help in ensuring that processes can only access the memory regions they are permitted to.

4. Mitigation of Common Attacks

Buffer Overflow Protection:

- Techniques such as bounds checking, stack canaries, and safe library functions help prevent buffer overflows, which can lead to arbitrary code execution or data corruption.

Heap Exploitation Protections:

- Strategies like heap canaries, safe unlinking, and using secure allocation libraries help protect against heap-based attacks.

Code Injection Prevention:

- Practices like code signing, secure coding techniques, and application whitelisting help mitigate risks associated with code injection attacks.

5. Secure Memory Allocation

Memory Sanitization:

- Ensure that sensitive information is properly erased from memory before it is reused or freed to prevent unauthorized access.

Secure Memory Allocation Libraries:

- Using libraries designed to handle memory securely, such as those with built-in protections against memory-related vulnerabilities.

6. Monitoring and Auditing

Memory Access Auditing:

- Implementing tools and techniques for monitoring memory access patterns can help detect unusual or unauthorized access attempts.

Regular Updates and Patches:

- Keeping the OS and its components updated is essential to protect against known vulnerabilities that could be exploited to compromise memory security.

Overall, memory security in operating systems involves a combination of hardware support, software techniques, and best practices aimed at protecting against various types of memory-related attacks and ensuring the integrity and confidentiality of data.

FILE

Protection in File System

In computer systems, a lot of user's information is stored, the objective of the operating system is to keep safe the data of the user from the improper access to the system. Protection can be provided in number of ways. For a single laptop system, we might provide protection by locking the computer in a desk drawer or file cabinet. For multi-user systems, different mechanisms are used for the protection.

Types of Access :

The files which have direct access of the any user have the need of protection. The files which are not accessible to other users doesn't require any kind of protection. The mechanism of the protection provide the facility of the controlled access by just limiting the types of access to the file. Access can be given or not given to any user depends on several factors, one of which is the type of access required.

Several different types of operations can be controlled:

- Read – Reading from a file.
- Write – Writing or rewriting the file.
- Execute – Loading the file and after loading the execution process starts.
- Append – Writing the new information to the already existing file, editing must be end at the end of the existing file.
- Delete – Deleting the file which is of no use and using its space for the another data.
- List – List the name and attributes of the file.
- Operations like renaming, editing the existing file, copying; these can also be controlled.

Access Control :

There are different methods used by different users to access any file. The general way of protection is to associate identity-dependent access with all the files and directories an list called access-control list (ACL) which specify the names of the users and the types of access associate with each of the user. The main problem with the access list is their length. If we want to allow everyone to read a file, we must list all the users with the read access. This technique has two undesirable consequences:

Constructing such a list may be tedious and unrewarding task, especially if we do not know in advance the list of the users in the system.

Previously, the entry of the any directory is of the fixed size but now it changes to the variable size which results in the complicates space management. These problems can be resolved by use of a condensed version of the access list. To condense the length of the access-control list, many systems recognize three classification of users in connection with each file:

Owner – Owner is the user who has created the file.

Group – A group is a set of members who has similar needs and they are sharing the same file.

Universe – In the system, all other users are under the category called universe.

The most common recent approach is to combine access-control lists with the normal general owner, group, and universe access control scheme. For example: Solaris uses the three categories of access by default but allows access-control lists to be added to specific files and directories when more fine-grained access control is desired.

Other Protection Approaches:

- The access to any system is also controlled by the password. If the use of password is random and it is changed often, this may be result in limit the effective access to a file.
- The use of passwords has a few disadvantages:
- The number of passwords are very large so it is difficult to remember the large passwords.
- If one password is used for all the files, then once it is discovered, all files are accessible; protection is on all-or-none basis.

TIME

Time is a crucial element in operating system security for a variety of reasons, including ensuring accurate event logging, managing access controls, and supporting cryptographic operations. Here's a detailed look at how time impacts security in operating systems and the measures taken to protect and manage it effectively:

1. Time Synchronization

Network Time Protocol (NTP):

- **Purpose:** Ensures that all systems in a network have a consistent time reference, which is vital for accurate logging, authentication, and other security functions.
- **Implementation:** Configure systems to synchronize with reliable NTP servers to prevent discrepancies that could lead to security issues or logging inaccuracies.

Precision Time Protocol (PTP):

- **Purpose:** Provides higher precision than NTP for environments that require very accurate time, such as financial systems or high-performance computing.
- **Implementation:** Use PTP in scenarios where extremely accurate timekeeping is critical, with proper configuration to minimize synchronization errors.

2. Secure Timekeeping

Time Source Integrity:

- **Purpose:** Protects the time source from tampering or manipulation that could impact system security.
- **Implementation:** Use secure and trusted time sources, employ redundancy (multiple time servers), and monitor for anomalies in time data.

Hardware Security Modules (HSMs):

- **Purpose:** Securely manage cryptographic keys and perform operations that involve time-sensitive data.
- **Implementation:** Utilize HSMs to ensure that cryptographic operations related to time (e.g., digital signatures with timestamps) are performed securely.

3. Time-Based Authentication

Time-Based One-Time Passwords (TOTP):

- **Purpose:** Enhances authentication security by generating passwords that change periodically.
- **Implementation:** Integrate TOTP into two-factor authentication (2FA) systems to provide an additional layer of security beyond static passwords.

Session Expiration:

- **Purpose:** Limits the duration of authentication sessions and tokens to reduce the risk of unauthorized access.
- **Implementation:** Set appropriate session timeouts and token expiration policies to ensure that access is revoked after a certain period or inactivity.

4. Logging and Auditing

Accurate Timestamps:

- **Purpose:** Provides precise timing for events recorded in logs, essential for auditing and forensic investigations.
- **Implementation:** Synchronize system clocks and ensure that logs include accurate timestamps, often using NTP or PTP to maintain consistency.

Log Integrity:

- **Purpose:** Prevents tampering with logs to ensure they are reliable and trustworthy for security audits and investigations.
- **Implementation:** Use write-once, append-only logging mechanisms, secure logs with encryption, and implement access controls to protect log data.

5. Time-Based Access Control

Scheduled Access Restrictions:

- **Purpose:** Restricts access to resources based on time of day or other time-related criteria.
- **Implementation:** Configure systems to enforce access control policies that limit user access to certain hours or based on scheduled maintenance windows.

Time-Based Policy Enforcement:

- **Purpose:** Enforces security policies that are dependent on time conditions.
- **Implementation:** Set policies for account locking after inactivity, enforce periodic password changes, and implement timed access restrictions.

6. Anti-Replay Protection

Timestamping in Protocols:

- **Purpose:** Prevents replay attacks by including timestamps in communications to ensure they are processed within a valid time window.
- **Implementation:** Include time-based tokens or timestamps in security protocols to validate the freshness of requests or messages.

Nonce Usage:

- **Purpose:** Provides an additional layer of protection against replay attacks by ensuring that each request or transaction is unique.
- **Implementation:** Use nonces (unique, single-use numbers) along with timestamps to protect against replay attacks in authentication and transaction protocols.

7. Compliance and Best Practices

Regulatory Compliance:

- **Purpose:** Adheres to regulations and standards requiring accurate timekeeping and logging for compliance purposes.
- **Implementation:** Follow industry standards and guidelines (e.g., PCI-DSS, HIPAA) for time management, logging accuracy, and security practices.

Regular Time Reviews:

- **Purpose:** Ensures that time management practices remain effective and adapt to changing security needs.
- **Implementation:** Periodically review and update time synchronization configurations, logging practices, and time-based policies.

OBJECT PROTECTION REQUIREMENTS AND TECHNIQUES IN OPERATING SYSTEM

In the context of operating systems (OS), object protection involves safeguarding various system resources such as files, processes, memory, and hardware. Here's a detailed look at the requirements and techniques for object protection within an OS:

1. File System Protection

Requirements:

- **Access Control:** Define who can read, write, execute, or modify files and directories.
- **Integrity:** Ensure that files cannot be tampered with or altered by unauthorized users.

Techniques:

- **File Permissions:** Implement user-based or role-based permissions (read, write, execute) for files and directories.
- **Access Control Lists (ACLs):** Use ACLs to define more granular access rights for users and groups.
- **Encryption:** Encrypt sensitive files to protect their content from unauthorized access.

2. Process Protection

Requirements:

- **Isolation:** Prevent processes from interfering with each other or accessing unauthorized resources.
- **Privilege Separation:** Restrict processes to the minimum necessary privileges to limit potential damage from vulnerabilities.

Techniques:

- **Process Isolation:** Use process isolation techniques (e.g., separate address spaces) to prevent one process from accessing the memory of another.
- **Privilege Levels:** Utilize different privilege levels (e.g., user mode and kernel mode) to restrict what processes can do.
- **Sandboxing:** Run untrusted code in a sandbox environment to limit its access to system resources.

3. Memory Protection

Requirements:

- **Segmentation:** Prevent processes from accessing or modifying the memory allocated to other processes.
- **Protection from Exploits:** Guard against buffer overflow attacks and other memory-related exploits.

Techniques:

- **Virtual Memory:** Implement virtual memory management to give each process its own address space.
- **Memory Segmentation and Paging:** Use segmentation and paging techniques to control memory access and ensure separation between processes.
- **Non-Executable Memory:** Mark certain areas of memory (e.g., stack and heap) as non-executable to prevent execution of malicious code.

4. Authentication and Authorization

Requirements:

- **User Identification:** Verify the identity of users trying to access the system.
- **Access Control:** Ensure users can only access resources and perform actions for which they have explicit permission.

Techniques:

- **Authentication Methods:** Implement authentication mechanisms like passwords, multi-factor authentication (MFA), biometrics, or smart cards.
- **Authorization Models:** Use access control models (e.g., discretionary access control (DAC), mandatory access control (MAC), or role-based access control (RBAC)) to enforce access policies.

5. System Integrity

Requirements:

- **Integrity Checking:** Ensure that the OS and its components are not tampered with or corrupted.

- **Patch Management:** Keep the system updated with the latest security patches and updates.

Techniques:

- **Checksums and Hashes:** Use checksums and cryptographic hashes to verify the integrity of system files and applications.
- **File Integrity Monitoring:** Implement file integrity monitoring tools to detect unauthorized changes to critical system files.
- **Patch Management Systems:** Regularly update the OS and applications with security patches and updates.

6. Auditing and Logging

Requirements:

- **Activity Tracking:** Keep track of system activity to detect and respond to security incidents.
- **Forensics:** Collect and preserve evidence for forensic analysis in case of a security breach.

Techniques:

- **System Logs:** Configure logging for system events, user activity, and security incidents.
- **Audit Trails:** Maintain audit trails to track changes to system configurations and access to sensitive data.
- **Log Analysis:** Use log analysis tools to monitor for unusual activity and potential security breaches.

7. Network Protection

Requirements:

- **Network Security:** Safeguard communication between processes, users, and external networks.
- **Data Privacy:** Protect data transmitted over the network from eavesdropping and tampering.

Techniques:

- **Firewalls:** Implement firewalls to control incoming and outgoing network traffic.
- **Intrusion Detection Systems (IDS):** Use IDS to monitor network traffic for suspicious activity.
- **Network Encryption:** Encrypt data transmitted over the network using protocols like TLS/SSL to protect data in transit.

By implementing these requirements and techniques, operating systems can better protect their resources and maintain overall system security.

IDENTIFICATION AND AUTHENTICATION IN OS

Identification and authentication are critical components of security in operating systems (OS). They ensure that only authorized users and processes can access system resources and perform specific actions. Here's a detailed look at how identification and authentication work in an OS:

Identification

1. Definition:

- **Identification** is the process of recognizing or stating who someone or something is. In the context of an OS, it typically involves a user or process claiming an identity to the system.

2. Components:

- **Usernames:** The most common method of identification. Each user has a unique username that the OS uses to distinguish between different users.
- **Process IDs:** For processes, the OS assigns unique Process IDs (PIDs) to identify each running process.

3. Methods:

- **User Accounts:** Each user is associated with an account that includes a username and potentially other attributes (e.g., group memberships, permissions).
- **System Attributes:** Processes and services might be identified based on attributes such as their PID or service name.

Authentication

1. Definition:

- **Authentication** is the process of verifying the identity claimed during identification. It ensures that the user or process is who they say they are.

2. Methods of Authentication:

- **1. Password-Based Authentication:**
  - **Description:** The user provides a username and a password. The OS verifies the password against stored credentials.
  - **Strengths:** Simple and widely used.
  - **Weaknesses:** Susceptible to password guessing, brute force attacks, and phishing.
- **2. Multi-Factor Authentication (MFA):**
  - **Description:** Requires multiple forms of verification (e.g., something you know, something you have, something you are).
  - **Examples:** Combination of a password, a smartphone authentication app, and a fingerprint scan.
  - **Strengths:** Provides higher security by adding additional verification steps.
  - **Weaknesses:** Can be more complex to set up and use.
- **3. Biometric Authentication:**
  - **Description:** Uses unique biological characteristics for verification (e.g., fingerprints, facial recognition, iris scans).
  - **Strengths:** Difficult to spoof and offers a convenient user experience.

    - **Weaknesses:** Biometric data can be stolen or spoofed; privacy concerns.
- **4. Smart Cards and Tokens:**
    - **Description:** Physical devices (e.g., smart cards, USB tokens) that generate or store authentication data.
    - **Strengths:** Adds a physical element to authentication, enhancing security.
    - **Weaknesses:** Physical tokens can be lost or stolen; require hardware support.
- **5. Public Key Infrastructure (PKI):**
    - **Description:** Uses asymmetric encryption (public and private keys) for authentication.
    - **Examples:** Digital certificates and signatures.
    - **Strengths:** Provides strong security for communications and data integrity.
    - **Weaknesses:** Can be complex to implement and manage.
- **6. Single Sign-On (SSO):**
    - **Description:** Allows users to authenticate once and gain access to multiple systems or applications.
    - **Strengths:** Simplifies user experience and reduces password fatigue.
    - **Weaknesses:** If the SSO system is compromised, multiple services may be at risk.

3. Implementation in OS:

- **1. Authentication Services:**
    - **Description:** OSs use authentication services to handle the process of verifying user credentials. These might include local services or integrations with centralized authentication systems like LDAP or Active Directory.
- **2. Credential Storage:**
    - **Description:** Credentials (e.g., hashed passwords, encryption keys) are stored securely, often in hashed or encrypted forms, to prevent unauthorized access.
    - **Techniques:** Passwords are typically hashed using cryptographic hashing algorithms (e.g., SHA-256) and may be salted to prevent rainbow table attacks.
- **3. Session Management:**
    - **Description:** Once authenticated, users are granted access and a session is created. The OS manages user sessions, keeping track of permissions and activity.
    - **Techniques:** Sessions are often managed through session tokens or cookies, and may include timeout features to enhance security.
- **4. Access Control Integration:**
    - **Description:** Authentication is integrated with access control mechanisms to enforce policies about what authenticated users can do.
    - **Techniques:** The OS uses access control lists (ACLs), role-based access control (RBAC), or other models to determine access based on authentication.

By effectively managing identification and authentication, operating systems can significantly enhance their security posture, ensuring that only authorized users and processes gain access to sensitive resources.

TRUSTED OPERATING SYSTEM

A Trusted Operating System (TOS) is designed with a focus on providing a high level of security and trustworthiness. Such systems are often used in environments where security and integrity are paramount, such as in military, financial, or other sensitive sectors. Here's a detailed overview of what constitutes a Trusted Operating System:

Key Characteristics of a Trusted Operating System

1. **Security Policies and Mechanisms:**
    - **Mandatory Access Control (MAC):** Unlike Discretionary Access Control (DAC), which allows users to set permissions on their files, MAC enforces security policies that cannot be altered by users. It ensures that access decisions are made based on predefined policies.
    - **Role-Based Access Control (RBAC):** Access to resources is granted based on the roles assigned to users, rather than individual permissions. This simplifies management and enforcement of security policies.
2. **Enhanced Access Control:**
    - **Separation of Duties:** Ensures that critical tasks are divided among multiple users to prevent abuse of privileges. For example, the tasks of approving and auditing transactions are separated.
    - **Least Privilege Principle:** Users and processes are given the minimum level of access necessary to perform their tasks, reducing the risk of accidental or malicious damage.
3. **Auditing and Logging:**
    - **Comprehensive Auditing:** Trusted Operating Systems provide detailed logging of system activities, including access attempts, configuration changes, and security-related events. This helps in monitoring, auditing, and forensic analysis.
    - **Tamper-Evident Logs:** Logs are protected to prevent unauthorized modification or deletion, ensuring their integrity and reliability.
4. **Secure Communication:**
    - **Encryption:** Trusted OSs often use strong encryption protocols to protect data in transit and at rest, ensuring confidentiality and integrity.
    - **Secure Channels:** Communication between processes or between systems is secured to prevent interception and tampering.
5. **Integrity and Availability:**
    - **System Integrity:** Mechanisms are in place to ensure that the OS and its components are not tampered with. This might include checksums, digital signatures, and secure boot processes.
    - **High Availability:** Ensures that the system remains operational and accessible even in the event of hardware or software failures.
6. **Isolation and Separation:**
    - **Process Isolation:** Ensures that processes run in isolated memory spaces, preventing one process from affecting others.
    - **Virtualization and Containers:** Use virtualization or containerization to create isolated environments for different applications or users.
7. **Trusted Computing Base (TCB):**
    - **Definition:** The TCB encompasses all hardware, firmware, and software components that are critical to enforcing security policies and maintaining the system's integrity.
    - **Evaluation:** Trusted OSs often undergo rigorous evaluation and certification processes to ensure that their TCB meets security standards.
8. **Compliance with Standards:**
    - **Common Criteria (CC):** Many Trusted Operating Systems are evaluated against the Common Criteria for Information Technology Security Evaluation (ISO/IEC 15408) to ensure they meet specified security requirements.
    - **Federal Information Processing Standards (FIPS):** Compliance with FIPS, such as FIPS 140-2 for cryptographic modules, is common in environments requiring strict security measures.

Examples of Trusted Operating Systems

1. **SELinux (Security-Enhanced Linux):**
   - **Description:** An extension of the Linux kernel that provides a robust framework for implementing MAC policies. It is used to enforce security policies and control access based on policies defined by system administrators.
   - **Usage:** Often used in environments requiring stringent security controls, including government and military applications.
2. **Trusted Solaris:**
   - **Description:** An operating system developed by Sun Microsystems (now Oracle) that includes enhanced security features such as MAC and RBAC.
   - **Usage:** Used in environments where high security and reliability are required.
3. **QNX Neutrino:**
   - **Description:** A real-time operating system designed for embedded systems with a focus on reliability, security, and real-time performance.
   - **Usage:** Commonly used in automotive, medical devices, and industrial control systems.
4. **Windows Server with Enhanced Security Features:**
   - **Description:** Microsoft Windows Server includes various security features and configurations that align with Trusted Operating System principles, such as BitLocker, AppLocker, and advanced auditing.
   - **Usage:** Widely used in enterprise environments where security and compliance are critical.
5. **Red Hat Enterprise Linux (RHEL) with Security-Enhanced Linux (SELinux):**
   - **Description:** A version of Linux that includes SELinux, providing enhanced security controls and policies.
   - **Usage:** Used in enterprise environments requiring robust security features.

UNIT – IV

DATABASE MANAGEMENT SYSTEMS SECURITY: Database integrity, Database secrecy, Inference control , Multilevel databases.

Introduction to Database Security (DBMS Security)

Database Security is a crucial aspect of any Database Management System (DBMS). It involves the implementation of measures to protect the database from unauthorized access, malicious threats, and vulnerabilities that could lead to data breaches or data loss. Ensuring the confidentiality, integrity, and availability of data are the main objectives of database security.

DATABASE INTEGRITY

Definition and Importance

Database Integrity refers to the accuracy, consistency, and reliability of data throughout its lifecycle, from the point of creation to deletion. Ensuring data integrity means maintaining the validity and correctness of data in a database, preventing it from becoming corrupted, incomplete, or inaccurate due to errors, unintended modifications, or malicious attacks.

Why is Database Integrity Important?

Accurate Decision Making: Organizations rely on data to make critical business decisions. Compromised data can lead to poor decision-making and strategic errors.

Compliance and Regulations: Maintaining data integrity is often required by industry regulations and standards such as GDPR, HIPAA, or SOX.

Trust and Reliability: Users, customers, and stakeholders must be able to trust the data within the system.

Data Security: Preserving data integrity is a key aspect of overall data security, as it ensures that data has not been altered or tampered with.

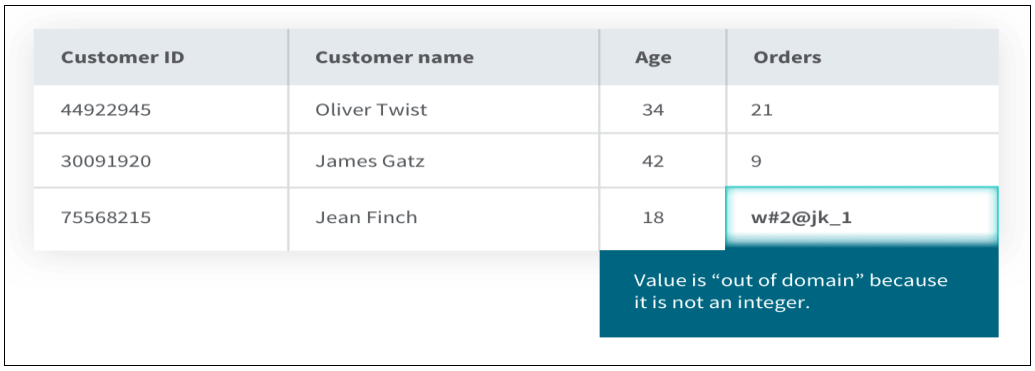
TYPES OF DATA INTEGRITY

To ensure integrity in both hierarchical and relational databases, there are the two primary types– physical integrity and logical integrity. Within logical integrity, there are four sub-categories: domain, entity, referential, and user-defined integrity. All are collections of rules and procedures which application programmers, system programmers, data processing managers, and internal auditors use to ensure accurate data.

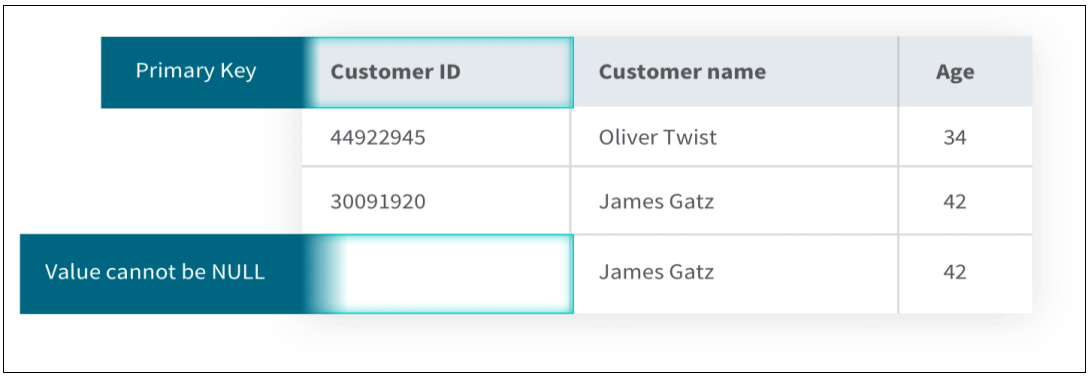
PHYSICAL INTEGRITY refers to the rules and procedures which ensure the accuracy of data as it is stored and retrieved. Threats to physical integrity include external factors such as power outages, natural disasters and hackers and internal factors such as storage erosion, human error or design flaws. Typically, the affected dataset is unusable.

LOGICAL INTEGRITY seeks to ensure that the data accurately makes sense in a specific context (whether it's "logical"). Logical integrity also has the challenge of human errors and design flaws. Thankfully, a dataset can be overwritten with new data and reused if it has a logical error. There are four topics of logical integrity as follows:

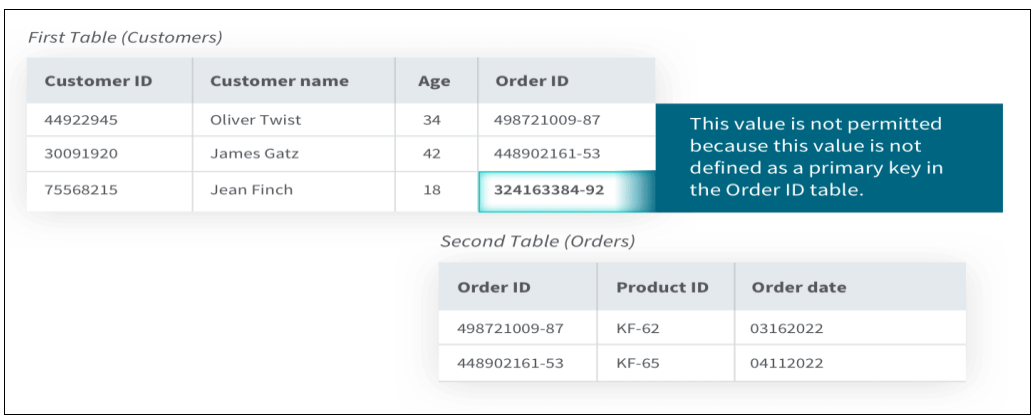
1. **Domain integrity** refers to the range of values such as integer, text, or date which are acceptable to be stored in a particular column in a database. This set of values (the "domain") has constraints which limit the format, amount, and types of data entered. All entries must be available in the domain of the data type. As shown in the example below, the entry for the number of Jean's orders is not an integer so it is out of domain. This would cause the database management system to produce an error.



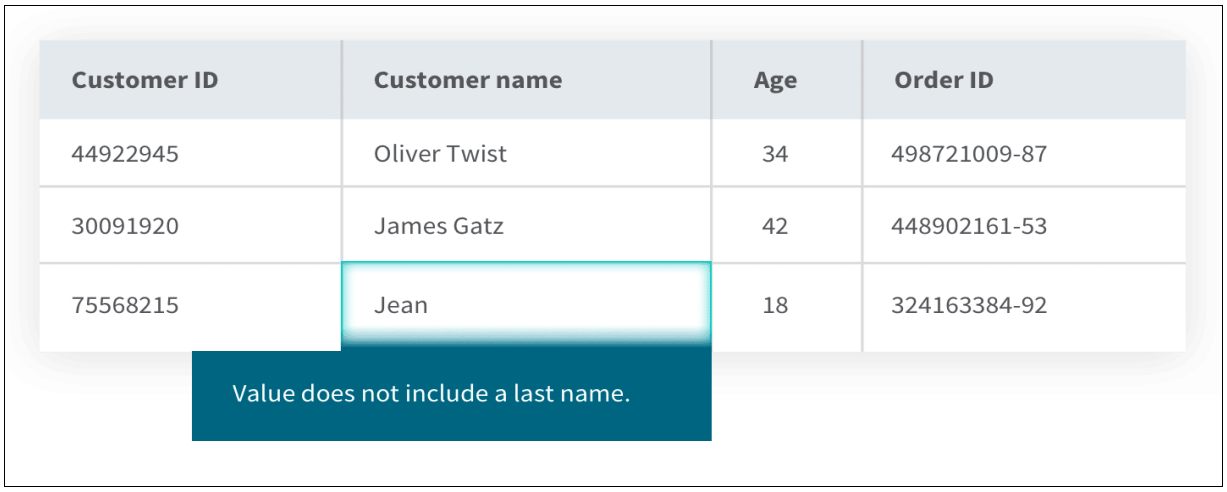
2. Entity integrity uses primary keys to uniquely identify records saved in a table in a relational database. This prevents them from being duplicated. It also means they can't be NULL because then you couldn't uniquely identify the row if the other fields in the rows are the same. For example, you might have two customers with the same name and age, but without the unique identifier of the customer ID primary key, you could have errors or confusion when pulling the data.



2. **Referential integrity** refers to the collection of rules and procedures used to maintain data consistency between two tables. These rules are embedded into the database structure regarding how foreign keys can be used to ensure the data entry is accurate, there is no duplicate data, and, as in the example below, data which doesn't apply is not entered. You can see below how referential integrity is maintained by not allowing an order ID which does not exist in the order table.



3. **User-defined integrity** acts as a way to catch errors which domain, referential and entity integrity do not. Here, you define your own specific business rules and constraints which trigger automatically when predefined events occur. For instance, you could define the constraint that customers must reside in a certain country to be entered into the database. Or, as in the example below, you might require that customers provide both first and last names.



MECHANISMS TO ENSURE DATABASE INTEGRITY

Constraints: Rules applied to columns or tables that enforce data integrity.

Examples include:

Primary Key: Ensures that each record is unique and non-null.

Foreign Key: Ensures that a value in one table corresponds to a value in another table.

Unique: Ensures that all values in a column are unique.

Check: Ensures that a value in a column meets a specified condition.

Not Null: Ensures that a column cannot contain null values.

Transactions and ACID Properties: Databases use transactions and ACID properties (Atomicity, Consistency, Isolation, Durability) to maintain data integrity:

Atomicity: Ensures that all operations within a transaction are completed successfully or none are executed.

Consistency: Ensures that the database moves from one consistent state to another.

Isolation: Ensures that transactions are executed in isolation, preventing interference between concurrent transactions.

Durability: Ensures that once a transaction is committed, it remains permanently in the system.

Triggers: Automatically execute predefined actions in response to specific changes or events in a table. For example, a trigger might check for business rule violations whenever a new row is inserted.

Stored Procedures: Encapsulated logic within the database that can enforce business rules and data integrity at a higher level.

Data Validation: Processes and checks implemented within the application layer to ensure that data being entered into the database meets specific requirements.

Backup and Recovery: Ensures that the data can be restored to its previous state in case of accidental deletion, corruption, or failure.

THREATS TO DATA INTEGRITY

Human Errors: Mistakes made by users during data entry, modification, or deletion.

Malware and Cyber Attacks: Malicious software or attackers can alter, delete, or corrupt data.

System Failures: Hardware malfunctions, power outages, or software bugs that cause data loss or corruption.

Concurrent Access Issues: Simultaneous transactions on the same data can lead to inconsistencies if not properly managed.

Poor Database Design: Inadequate design can result in data redundancy, anomalies, or inconsistencies.

BEST PRACTICES TO MAINTAIN DATA INTEGRITY

Design the Database with Integrity: Use appropriate primary keys, foreign keys, and constraints from the start.

Use Transactions: Implement transactions for any data modification operations to ensure that data remains consistent.

Regular Auditing and Monitoring: Implement logs and auditing mechanisms to monitor changes and detect anomalies.

Data Validation and Cleansing: Validate data at multiple levels (application and database) and perform regular data cleansing to remove inconsistencies.

Implement Backup and Disaster Recovery Plans: Regular backups and tested recovery procedures help ensure data can be restored in case of corruption or loss.

Use Integrity Constraints and Triggers: Apply constraints and triggers to automate the enforcement of data integrity rules.

DATABASE SECRECY IN DBMS

Database secrecy in a Database Management System (DBMS) refers to protecting sensitive data from unauthorized access and ensuring that only legitimate users can view or manipulate the information. It's a crucial aspect of database security and involves several strategies and techniques designed to maintain the confidentiality of stored information.

Key Concepts of Database Secrecy

1.Access Control Mechanisms:

Authentication: Ensuring that only authenticated users can access the database. Authentication can be enforced using usernames, passwords, biometrics, or multi-factor authentication.

Authorization: Granting permissions to users to access only the data they are supposed to. This can be done using role-based access control (RBAC), where permissions are assigned based on roles, or through discretionary access control (DAC), where the data owner grants permissions directly.

2. Data Encryption:

Data is often encrypted both at rest (when stored in the database) and in transit (when being transmitted over networks). This ensures that even if someone gains unauthorized access, they cannot interpret the data without the decryption keys.

3.Database Views:

Views can be used to restrict access to certain parts of the database. Instead of providing access to entire tables, views allow users to see only a subset of data that they have permissions to access, enhancing secrecy by obscuring sensitive information.

4.Fine-Grained Access Control (FGAC):

FGAC involves implementing access control at a very detailed level, such as at the row or column level. This allows administrators to restrict access to specific pieces of information within a table, providing more control over what users can see.

5.Label-Based Security:

This method assigns security labels to data and user accounts. Data labels classify the sensitivity of the data, and user labels define what classification levels they are authorized to access. This is often used in government or military applications for classified information.

6.Audit Trails and Monitoring:

Tracking who accesses or manipulates the database is essential for maintaining secrecy. Auditing ensures that unauthorized access attempts or suspicious activities are recorded, allowing for timely detection and response to potential security breaches.

7.Physical Security:

Securing the physical hardware that hosts the database is just as crucial as securing the digital data. Unauthorized physical access to servers can lead to a breach of data secrecy.

8.Database Anonymization and Masking:

These techniques are used to hide sensitive information by transforming the data in such a way that it cannot be traced back to the original value. Data masking replaces sensitive data with fictional data, while anonymization removes personally identifiable information (PII) to prevent linking back to an individual.

Implementing Database Secrecy in Practice

To implement database secrecy effectively, a combination of the above techniques is usually employed. For example, a healthcare organization might use:

Authentication to ensure only doctors and administrative staff can access the database.

Authorization to allow doctors to view medical records while restricting administrative staff to only see billing information.

Data encryption to secure medical records both at rest and during transmission.

Audit trails to log access attempts for regulatory compliance.

By combining these strategies, the organization ensures that sensitive patient information remains confidential and secure.

Challenges in Database Secrecy

Managing and Updating Permissions: As the database grows and new users join or leave, keeping permissions up to date can become complex and error-prone.

Data Integrity vs. Secrecy: While ensuring secrecy, it's also crucial to maintain data integrity and availability, which sometimes creates trade-offs.

Performance Impact: Encryption, access control, and auditing can impose additional overhead on database performance, making it a challenge to balance security and efficiency.

Overall, maintaining database secrecy is a dynamic and ongoing process, requiring consistent updates and monitoring to address evolving threats and vulnerabilities.

INFERENCE CONTROL IN DBMS SECURITY

Inference control in DBMS security refers to techniques and methods used to prevent users from deriving or inferring unauthorized information from legitimate queries or transactions. This is a crucial aspect of database security, especially in multi-user environments, where different users may have access to different levels of information. The goal of inference control is to ensure that sensitive data remains confidential, even when users are allowed to access or query less sensitive parts of the database.

What is Inference?

Inference refers to the process of deducing sensitive or restricted information by analyzing accessible data. Inference attacks occur when a user or an attacker uses non-sensitive data and logical reasoning to gain insight into sensitive data.

For example:

If a user is allowed to query average salaries in a department, they might infer an individual's salary by subtracting the known salaries from the total sum.

If a user knows that a specific person was recently diagnosed with a disease and can query the hospital's database for patient counts by disease type, they might infer sensitive health information about that individual.

Why is Inference Control Important?

Inference attacks are difficult to detect because users aren't directly accessing restricted data—they are deducing it indirectly. These attacks can undermine database security policies, leading to the exposure of confidential information and violating data privacy regulations.

Therefore, DBMSs must implement inference control to prevent unauthorized access through deduction.

Techniques for Inference Control

Query Restriction:

Suppression: Suppressing certain query results if they can lead to inference of sensitive information. This means that the DBMS denies the execution of certain queries that could reveal confidential data.

Concealing Results: Modifying query results to conceal or mask sensitive information. This might involve not showing certain fields or replacing sensitive data with generic values.

Data Perturbation:

Perturbation involves adding noise or minor modifications to the data or query results, such as slightly altering numerical values or introducing random values in the result set. This technique ensures that users cannot make exact inferences based on the information they receive.

Partitioning and Aggregation:

Dividing the data into partitions that cannot be individually used to infer sensitive information. Aggregated data is presented in such a way that individual records cannot be distinguished or inferred.

Query History Tracking and Analysis:

Keeping track of all queries submitted by a user and analyzing these queries over time to detect patterns that might lead to inference. If a user's query history suggests they are trying to infer confidential information, restrictions can be applied.

Randomized Response:

A technique commonly used in statistical databases where some of the answers to queries are randomly changed to introduce uncertainty. This technique prevents users from accurately determining specific values.

Multi-Level Security (MLS):

Multi-Level Security mechanisms assign security labels to both users and data (e.g., “Confidential,” “Secret,” “Top Secret”). Access is controlled based on these labels, and inference control ensures that users cannot deduce higher-level sensitive information by querying lower-level non-sensitive data.

Integrity Constraints and Rules:

Implementing integrity constraints and rules within the database schema to enforce restrictions on query results. For example, certain rows or columns can be restricted if accessing them would lead to a violation of inference control policies.

Data Dependency Analysis:

Analyzing how data items in a database are related to one another. The DBMS can limit query results if it detects that relationships between certain data items might allow a user to infer confidential information.

Example of Inference Attacks and Controls

Example 1: A user queries the average salary of employees in a department and then queries the total salary of all employees. By subtracting the known salaries of a few employees, they might infer the salary of a specific individual.

Control: Implement query restrictions that prevent users from querying both the average salary and the total salary in the same session or for the same group.

Example 2: A healthcare database allows users to query the number of patients diagnosed with a certain disease. If a user knows that only a single patient has been recently admitted, they can infer that this person has the disease.

Control: Apply data perturbation by slightly modifying the patient counts or providing result ranges instead of exact values to prevent disclosure of exact counts.

Challenges in Inference Control

Performance Overhead: Implementing inference controls, especially those involving query tracking or data perturbation, can introduce additional processing overhead and slow down database performance.

Complexity of Detection: Detecting and mitigating all possible inference channels is extremely challenging, as it requires comprehensive knowledge of data relationships and user intentions.

Balancing Security and Usability: Implementing too strict inference controls can negatively impact the usability of the database, making it hard for legitimate users to perform their tasks.

A successful inference control mechanism not only protects data but also maintains a balance between security, performance, and usability.

MULTILEVEL DATABASE

A multilevel database is a type of database architecture designed to handle data classified at multiple security levels. It is typically used in environments with stringent security requirements, such as military, government, and financial institutions, where different users have varying permissions to access data based on their security clearance levels.

What is a Multilevel Database?

In a multilevel database system (MLDB), each data item is assigned a security classification label, and each user is assigned a security clearance level. These labels and clearances dictate the level of access each user has to specific data. This concept ensures that users can only access information for which they have the necessary clearance, thus enforcing strict data confidentiality and preventing unauthorized access or disclosure.

Characteristics of Multilevel Databases

1. **Data Classification Levels**:

    - Each piece of data in a multilevel database is assigned a classification level, such as "Top Secret," "Secret," "Confidential," or "Unclassified." These labels represent different levels of sensitivity and dictate which users can access them.
2. **User Security Levels**:
    - Every user is assigned a security clearance level that corresponds to the data classification levels. This clearance level determines the maximum level of sensitivity they are authorized to access.
3. **Access Control Mechanism**:
    - The access control mechanism ensures that users can only read, write, or update data that they are authorized to access. This is typically enforced using rules derived from the Bell-LaPadula model (for confidentiality) and the Biba model (for integrity).
4. **Multi-Level Security (MLS) Policies**:
    - The database adheres to MLS policies, such as "no read-up" (users cannot read data at a higher security level than their clearance) and "no write-down" (users cannot write data to a lower security level, to avoid leaking information).

Multilevel Database Models

Multilevel databases rely on formal security models to enforce data access rules. Some of the key models used are:

1. **Bell-LaPadula Model (BLP)**:
    - This model focuses on data confidentiality. The key policies are:
        - **No Read-Up (Simple Security Property)**: Users cannot read data at a higher security level than their clearance.
        - **No Write-Down (Star Property)**: Users cannot write data at a lower security level than their clearance.
2. **Biba Model**:
    - This model focuses on data integrity. The key policies are:
        - **No Write-Up**: Users cannot write data at a higher security level to prevent corrupting sensitive data.
        - **No Read-Down**: Users cannot read data at a lower security level, ensuring they only access information that has been validated as appropriate for their security level.
3. **Clark-Wilson Model**:
    - This model is used to enforce well-formed transactions and separation of duties, which helps in ensuring data consistency and integrity within a multilevel database system.

Implementation Strategies for Multilevel Databases

There are several strategies for implementing multilevel databases:

1. **Single-Level Database with Multiple Views**:
    - A traditional single-level database with access controls and views can be used to simulate a multilevel database. Each user has access to specific views based on their clearance level.
2. **Polyinstantiation**:
    - Polyinstantiation allows the storage of multiple instances of the same data item, each with different security classifications. This prevents inference attacks by ensuring that each user sees only the data they are cleared to see, without indicating the existence of other classified data.
3. **Data Partitioning**:

    - Data can be physically or logically partitioned into separate tables or databases, each corresponding to a different security level. Users can access only those partitions for which they have clearance.
4. **Label-Based Security**:
    - Data items are tagged with labels that indicate their classification level, and user sessions are associated with security labels. Access is controlled by comparing the user's label to the data label.

Advantages and Challenges of Multilevel Databases

Advantages:

- **Enhanced Security**: Multilevel databases provide robust security controls that prevent unauthorized access and ensure data confidentiality.
- **Prevents Data Leakage**: By enforcing policies like "no read-up" and "no write-down," MLDBs prevent users from inferring or disclosing sensitive data beyond their clearance.
- **Supports Classified Environments**: Suitable for environments with stringent security requirements, such as military and government agencies.

Challenges:

- **Complexity in Management**: Maintaining and enforcing access controls for multiple levels of security can be complex and prone to errors.
- **Performance Overhead**: Due to the additional access controls and security mechanisms, MLDBs may experience performance issues compared to single-level databases.
- **Polyinstantiation Handling**: Managing multiple versions of the same data item (polyinstantiation) can introduce storage and data consistency challenges.
- **Inference Control**: Preventing inference attacks in multilevel databases is a significant challenge and requires sophisticated controls.

Example Scenario

Consider a multilevel database used in a military organization:

- **Data Classification Levels**:
    - **Top Secret**: Contains high-level military strategies and operational plans.
    - **Secret**: Contains mission details, equipment status, and personnel information.
    - **Confidential**: Contains general personnel details and logistical information.
    - **Unclassified**: Contains non-sensitive administrative information.
- **User Security Levels**:
    - **General (Top Secret Clearance)**: Can access all data, including Top Secret.
    - **Officer (Secret Clearance)**: Can access data classified as Secret or lower.
    - **Clerk (Confidential Clearance)**: Can access only Confidential and Unclassified data.

In this scenario, access control policies ensure that a Clerk cannot view mission details (Secret data) and an Officer cannot update Top Secret strategies.

UNIT V - NETWORK SECURITY

Network Threats : Eavesdropping, Spoofing, Modification, Denial of Service – Introduction to Network Security Techniques: Firewall, Intrusion Detection System, Cyber crimes and control measures

NETWORK THREAT - INTRODUCTION

A network threat refers to any potential activity or event that could harm or interrupt the systems, applications and services operating on a network. These threats can compromise the security of the network by attacking its infrastructure with the primary target usually being information theft or service disruption.

Examples of network threats include malware attacks, phishing attacks, ransomware, denial of service (DoS) attacks, unauthorized access, and data breaches, among others. Network threats can be initiated intentionally by threat actors such as hackers or unintentionally via software vulnerabilities and user errors.

EAVESDROPPING

Eavesdropping can be defined as a type of man-in-the-middle attack in which an individual intercepts, deletes, or modifies data being transmitted in real time between two devices. A phone call, instant message, video chat, fax transmission, and so on are all examples of data. To explain this in layman's terms: Eavesdropping is listening in on other people's conversations without their knowledge. This type of attack is also known as sniffing or snooping.

Eavesdropping can also occur when you share data on an open network without secured or encrypted traffic. The data is transmitted over an open network, allowing an attacker to exploit and intercept a vulnerability using various methods such as transmission links, pickup devices, etc.

Once you connect to an unprotected and encrypted network, you may unknowingly provide sensitive information to an attacker, such as passwords, account numbers, PAN, aadhaar no., etc.

Eavesdropping Methods

Attackers can use a variety of techniques to eavesdrop. Let's review the various methods widely used to launch an eavesdropping attack.

Pickup Device

To eavesdrop on targets, attackers can use devices that collect sound or images and converts them into an electrical format, such as microphones and video cameras. This device should ideally draw power from the power sources in the target room, removing the requirement for the attacker to go into the room to charge up or consider replacing the device's batteries.

Transmission Link

For listening purposes, an attacker can tap the transmission link between a pickup device and the attacker's collector. The attacker can accomplish this through a radiofrequency transmission or a cable, including active or unused phone lines, electric cables, or ungrounded electrical pipelines.

Open Networks

Clients interacting on open networks without passwords and without encrypting data provide an ideal environment for attackers to listen in. This is one of the most effective ways hackers monitor user activity and eavesdrop on network communications.

Weak Passwords

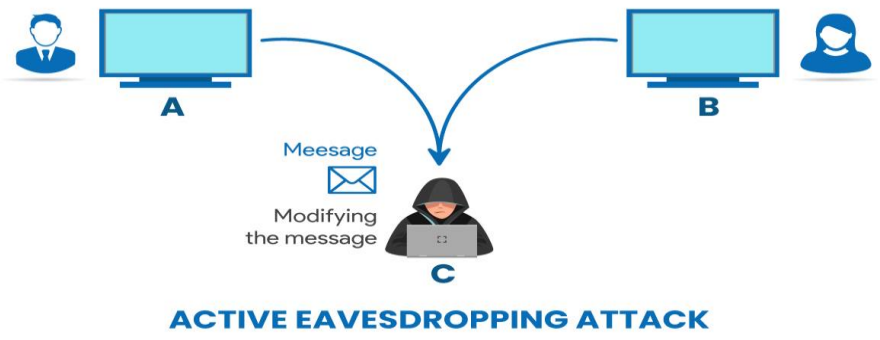
Weak passwords make it easier for hackers to gain unauthorized access to user accounts. Hackers use various attacks to gain login access, such as brute force attacks, social engineering attacks, etc. Once inside the system or network, hackers can easily infiltrate secret communication channels, intercept activity and conversations among coworkers, and steal sensitive information.

Types of Eavesdropping Attacks

Active eavesdropping and passive eavesdropping are the two types of eavesdropping attacks.

Active:

The attacker attempts to change the content of the messages in an active eavesdropping attack. Assume two people (A and B) are conversing on Telegram (a social media site). An attacker (C) will intercept the message sent by A and edit it as needed before forwarding it to person B. When B receives the message, he or she will be unaware of any tempering and assume that all of the message content is from person A.



Passive:

The attacker observes the content and may copy the content of the message, but no change is done to the content. Let's assume the above scenario. The only thing that will change will be the action of the attacker. Here he may copy the content of the message for later use, but no change is done to the content.

Real-life Example of Eavesdropping Attacks

A recent case occurred when a remote employee needed to send sensitive business information to his boss. However, because his home network was malfunctioning, he went to a nearby cafe and sent the information through their network. However, he was unaware that the network in the cafe was an open network. An attacker eavesdropped on the information, causing a massive loss for the organization.

What are the Impacts of Eavesdropping Attack?

Loss of privacy: Every business and person has private information that, if made public, can harm their reputation.

Identity theft: Attackers can listen in on any employee's private conversation to obtain login credentials, which they can then use to access restricted storage devices. Individuals do not simply lose their identities.

Financial loss: Attackers with sensitive data can access critical business applications anytime. They can threaten to reveal the information unless the victim pays a large sum or sells it to competitors.

How to Prevent Eavesdropping Attacks?

Use of VPN: A virtual private network (VPN) encrypts data between two points and is the most common form of eavesdropping protection. Corporate wireless networks should use the highest level of encryption possible.

Encryption: Encrypting data in transmission and private conversations is one of the best ways to prevent eavesdropping attacks. Encryption prevents attackers from reading data exchanged between two parties. Military-grade encryption is an excellent way to protect against eavesdropping attacks because it takes attackers 500 billion years to decode.

Avoid shady links: Eavesdropping attackers can spread malicious software that includes eavesdropping malware through shady links.

Keep your system up to date: Attackers can target organizations and users by exploiting vulnerabilities in old software. This makes it critical to update your system because it may include security patches.

Physical security: Organizations should also protect the data in their office spaces by implementing physical security measures. This is critical for protecting the office from unauthorized individuals who may place physical bugs on desks, phones, and other devices.

Set strong passwords and change them frequently: Use passwords that contain a combination of upper and lower case letters, numbers, and special characters. To improve security, you should change your passwords once a month.

Firewall: A personal firewall will protect your data packets from an intruder attempting to eavesdrop on your conversation.

Download software from official websites: Only download apps from trusted sources such as Google Play or Apple stores, as files downloaded from these platforms, will not be infected with malware, etc., that can download eavesdropping software without the user's permission.

SPOOFING

Spoofing in network security refers to a malicious attack where a person or program successfully masquerades as another by falsifying data to gain an illegitimate advantage. The main objective of spoofing is to deceive systems, users, or network components into thinking that they are interacting with a trusted source, when in fact, they are engaging with a malicious actor.

Common types of spoofing attacks include:

1. IP Spoofing: An attacker disguises their IP address to mimic a trusted source, allowing them to send and receive network traffic as if they were a legitimate user.

2. Email Spoofing: A fake sender address is used to trick recipients into believing that a message came from a legitimate source, often used in phishing attacks.

3. DNS Spoofing (or DNS Cache Poisoning): The attacker provides false DNS responses, directing users to malicious websites even though they think they are visiting legitimate ones.

4. ARP Spoofing: An attacker sends falsified ARP messages on a local network, linking their MAC address with the IP address of another host. This allows them to intercept, modify, or block traffic intended for the legitimate IP address.

5. Caller ID Spoofing: A technique where attackers alter the caller ID information to disguise their identity on phone systems, often used in phishing or vishing (voice phishing) scams.

Mitigation Techniques:

Authentication mechanisms: Use stronger, mutual authentication protocols to verify identities (e.g., IPsec, SSL/TLS).

Packet Filtering: Implement firewall and router rules to block spoofed traffic.

Encryption: Encrypt sensitive data to protect it from being intercepted and altered.

Security protocols: Use network security protocols such as DNSSEC to secure DNS communications, and secure routing protocols to prevent IP spoofing.

Spoofing can lead to unauthorized access, data theft, and network vulnerabilities, making it a significant concern in cybersecurity.

How can I protect against Spoofing Attacks?

For everyday users, the best way to protect against spoofing is by being vigilant for the signs of such an attack. As noted above, these include:

- **Never click unsolicited links** or download unexpected attachments.
- **Always log into your account through a new browser tab or official app** — not a link from an email or text.
- **Only access URLs that begin with HTTPS.**
- **Never share personal information**, such as identification numbers, account numbers or passwords, via phone or email.
- When contacted by a customer service representative via phone or email, **perform a Google search** to determine if the number or address is associated with any scams.

- **Use a password manager**, which will automatically enter a saved password into a recognized site (but not a spoofed site).
- **Use a spam filter** to prevent a majority of spoofed emails from reaching your inbox.
- **Invest in cybersecurity software**, which will detect many threats and even stop them from infecting your device.
- **Enable two-way authentication whenever possible**, which makes it far more difficult for attackers to exploit.

MODIFICATION IN NETWORK SECURITY

Modification in network security refers to unauthorized changes made to network traffic or data, often by malicious actors with the intent of altering the content or behavior of communication between systems. This type of attack can compromise data integrity, which is crucial for maintaining trust in a network.

Types of Modification Attacks:

1. Man-in-the-Middle (MITM) Attacks: In a MITM attack, the attacker intercepts and modifies the communication between two parties without their knowledge. They can alter the content of messages, inject malicious data, or change commands sent between users and systems.

2. Replay Attacks: In a replay attack, valid data transmission is maliciously or fraudulently repeated or delayed. Attackers capture legitimate data, such as login credentials, and resend it later to gain unauthorized access to the network or system.

3. Data Tampering: This involves altering data during transmission. Attackers may modify financial transactions, alter configuration settings, or change file content in transit, which can lead to data corruption or malicious actions being executed.

4. Session Hijacking: Attackers take over an active session by intercepting and modifying session tokens or cookies. This allows them to impersonate the legitimate user, often gaining unauthorized access to a network or system.

Consequences of Modification Attacks:

Loss of Data Integrity: Modified data may become inaccurate or corrupted, leading to erroneous decisions or actions based on altered information.

Loss of Confidentiality: Data might be exposed to unauthorized parties if intercepted and modified in transit.

Denial of Service: Modification attacks can lead to disruption of service if crucial network data or commands are altered.

Unauthorized Access: Attackers can gain unauthorized access by modifying data such as credentials or session tokens.

Mitigation Strategies:

1. Encryption: Encrypt data during transmission using secure protocols like TLS or IPSec. Encryption ensures that even if the data is intercepted, it cannot be easily altered or read by unauthorized parties.

2. Integrity Checks: Use hashing algorithms (e.g., SHA-256) to ensure that data has not been altered during transmission. The sender generates a hash of the data, and the receiver verifies that hash to detect any modifications.

3. Authentication: Implement strong authentication methods, such as multi-factor authentication (MFA), to ensure that the parties communicating are who they claim to be, reducing the risk of MITM attacks.

4. Digital Signatures: Digital signatures provide a way to verify the authenticity and integrity of data. If data is altered in transit, the signature will not match, alerting recipients to the modification.

5. Session Security: Secure sessions by using HTTPS, and regularly refresh session tokens to prevent hijacking and replay attacks.

DENIAL OF SERVICE (DOS)

In network security, **Denial of Service (DoS)** is a type of cyberattack where an attacker seeks to make a network, service, or system unavailable to its intended users. This is achieved by overwhelming the target with a flood of illegitimate requests, consuming its resources (such as bandwidth, memory, or CPU), or exploiting vulnerabilities, thus preventing legitimate traffic from accessing the service. DoS attacks can target websites, servers, or entire networks, causing downtime and disruption to normal operations.

Types of DoS Attacks:

1. **Flood Attacks**: The attacker sends an overwhelming volume of requests, which exceed the capacity of the targeted server or network, causing it to slow down or crash. Common flood attacks include:
    - **ICMP Flood (Ping Flood)**: Large volumes of ICMP (Internet Control Message Protocol) Echo Requests (pings) are sent, overwhelming the target.
    - **SYN Flood**: The attacker sends many SYN requests to establish a connection but never completes the TCP handshake, consuming server resources.
    - **UDP Flood**: A large number of UDP packets are sent to random ports on the target, forcing it to check for applications on each port and respond, consuming resources.
2. **Application Layer Attacks**: These attacks target specific applications, such as a web server or database, by overwhelming them with requests, making the service unavailable.
    - **HTTP Flood**: A flood of HTTP GET or POST requests is sent to a web server, causing it to exhaust its resources.
    - **Slowloris**: The attacker sends incomplete HTTP requests to the server, keeping connections open and eventually exhausting the server’s capacity.

3. **Distributed Denial of Service (DDoS)**: In a DDoS attack, multiple compromised systems (often forming a botnet) are used to target and overwhelm the victim. This makes DDoS attacks more challenging to mitigate compared to regular DoS, as the traffic originates from many sources, making it harder to block or trace back to the attacker.
4. **Amplification Attacks**: The attacker sends small requests to publicly accessible services that respond with much larger responses, amplifying the amount of traffic directed at the target. This can increase the impact of the attack significantly.
    - **DNS Amplification**: The attacker sends DNS queries to a public DNS resolver with a spoofed IP address (that of the target), causing the resolver to send large DNS responses to the target, overwhelming it.

Impact of DoS/DDoS Attacks:

- **Service Disruption**: The target service becomes slow or entirely unavailable to legitimate users.
- **Financial Loss**: Downtime may lead to lost revenue for businesses relying on online services.
- **Reputation Damage**: Repeated service outages can erode customer trust.
- **Increased Operational Costs**: Organizations may need to invest in mitigation tools and additional infrastructure to counter DoS attacks.

Mitigation Strategies:

1. **Rate Limiting**: Restricting the number of requests a system can handle in a given period, reducing the effectiveness of flood attacks.
2. **Firewalls and Intrusion Detection Systems**: Configured to filter malicious traffic and detect unusual patterns of activity.
3. **Load Balancers**: Distributing traffic across multiple servers can help prevent any single server from being overwhelmed.
4. **Anti-DDoS Solutions**: Dedicated services that detect and block malicious traffic in real time.
5. **Anycast Routing**: Distributing traffic across multiple data centers to absorb large-scale DDoS attacks by leveraging geographically dispersed infrastructure.

Denial of Service attacks are a serious threat in cybersecurity, often requiring a multi-layered defense strategy to protect against various types of attacks.

FIREWALL

A **firewall** is a network security device or software that monitors and controls incoming and outgoing network traffic based on predetermined security rules. Its primary purpose is to create a barrier between a trusted internal network and untrusted external networks, such as the internet, and prevent unauthorized access while permitting legitimate communications.

Types of Firewalls:

1. **Packet-Filtering Firewall**:
    - Operates at the network layer (Layer 3) and examines packets based on headers (source/destination IP addresses, ports, and protocols).
    - It allows or blocks packets based on predefined rules (Access Control Lists, or ACLs).
    - Simple and fast but lacks deep packet inspection, meaning it can't analyze the data within the packets.
2. **Stateful Inspection Firewall**:
    - Tracks the state of active connections and makes decisions based on the context of the traffic (i.e., whether a packet is part of an established, valid connection).
    - Offers more security than packet-filtering firewalls as it considers the connection state, source, and destination.
3. **Proxy Firewall (Application-Level Gateway)**:
    - Operates at the application layer (Layer 7) of the OSI model.
    - Acts as an intermediary between users and the external network by creating a separate connection to both the internal and external devices.
    - Filters traffic based on application-level protocols like HTTP, FTP, and DNS, providing more granular control.
    - Slower due to its deep packet inspection capability, but more secure.
4. **Next-Generation Firewall (NGFW)**:
    - Combines traditional firewall features with advanced functions like application awareness, deep packet inspection, intrusion prevention systems (IPS), and SSL decryption.
    - Can identify and control specific applications, even if they use non-standard ports or encryption, making them more effective against modern threats.
    - Incorporates threat intelligence and can prevent advanced threats like malware and zero-day attacks.

5. **Unified Threat Management (UTM) Firewall**:
    - An all-in-one security solution that combines firewall functionality with additional security features like anti-virus, content filtering, and VPN.
    - Common in small to medium-sized businesses where having a consolidated security system is cost-effective and easier to manage.

Firewall Architectures:

1. **Network-Based Firewalls**:
    - Deployed at the perimeter of the network, typically at the edge between an internal network and the external internet.
    - Protects the network as a whole by managing traffic between different networks (e.g., internal LAN and external internet).
2. **Host-Based Firewalls**:
    - Installed on individual devices (e.g., computers, servers, or mobile devices) to control traffic going in and out of that particular host.
    - Provides an additional layer of security, even when network-based firewalls are in place.

Key Firewall Functions:

1. **Traffic Filtering**: Blocking or allowing traffic based on defined security rules, such as IP addresses, protocols, and port numbers.
2. **Network Address Translation (NAT)**: Hides the internal IP addresses of a network behind a public IP address, which enhances security by masking the internal network.
3. **Logging and Monitoring**: Keeps track of all traffic and alerts administrators to suspicious activities or attempts to breach the network.
4. **Virtual Private Network (VPN) Support**: Often supports VPNs to allow secure, encrypted communication between remote users and the internal network.

Firewall Policies and Rules:

- Firewalls rely on **rule sets** to allow or block traffic. These rules are typically based on:
    - **Source IP address**: Where the traffic is coming from.
    - **Destination IP address**: Where the traffic is headed.
    - **Port numbers**: The service or application the traffic is using (e.g., HTTP on port 80, HTTPS on port 443).
    - **Protocols**: Rules can filter by protocol type (e.g., TCP, UDP, ICMP).
    - **Traffic direction**: Whether the traffic is inbound or outbound.

Benefits of Firewalls:

1. **Network Security**: Firewalls are a core component of network security, preventing unauthorized access and attacks.
2. **Traffic Management**: By filtering traffic, firewalls can optimize the flow of legitimate data and minimize unwanted or harmful connections.
3. **Access Control**: Firewalls can enforce access control policies, ensuring only trusted users or devices can interact with sensitive resources.

Limitations of Firewalls:

1. **Limited Protection**: While firewalls are essential, they cannot protect against all threats, especially if attacks come from within the network.
2. **Configuration Complexity**: Misconfigured firewalls can result in security gaps or overly restrictive rules that block legitimate traffic.
3. **Sophisticated Attacks**: Modern threats, such as encrypted attacks, zero-day exploits, or insider attacks, can sometimes bypass or evade firewall protections, requiring more advanced security solutions.

INTRUSION DETECTION SYSTEMS

An **Intrusion Detection System (IDS)** is a network security tool that monitors network traffic or system activities for suspicious behavior, anomalies, or known attack patterns, and alerts administrators of potential intrusions. Unlike firewalls, which actively block traffic based on predefined rules, an IDS is a **passive monitoring system** that does not take direct action to block potential threats but instead reports them, allowing for further investigation or response by security personnel.

Types of Intrusion Detection Systems:

1. **Network-based IDS (NIDS)**:
    - Monitors network traffic for signs of attacks or anomalies.
    - Typically deployed at strategic points within a network, such as at the boundary between the internal network and the internet, or in critical segments of the network.
    - Analyzes the data packets traveling over the network and looks for patterns that match known attack signatures or anomalous behavior.
2. **Host-based IDS (HIDS)**:
    - Monitors activity on individual hosts (e.g., computers, servers) for signs of suspicious behavior.
    - Analyzes logs, file integrity, and other system-related activities to detect potential intrusions.
    - Provides insight into threats that may have bypassed network defenses or originated from within the host itself (e.g., insider threats).

IDS Detection Techniques:

1. **Signature-Based Detection**:

    - The IDS uses a database of known attack signatures or patterns to identify potential threats.
    - Works similarly to antivirus software, where the system compares incoming traffic or system behavior against a database of known malicious patterns.
    - **Pros**: Highly accurate in detecting known attacks.
    - **Cons**: Cannot detect new or unknown threats (zero-day attacks) that do not match existing signatures.
2. **Anomaly-Based Detection**:
    - The IDS establishes a baseline of normal network or system behavior and then monitors for deviations from that baseline.
    - Any unusual activity (e.g., abnormal traffic volume, unusual user behavior) triggers an alert.
    - **Pros**: Can detect previously unknown attacks and zero-day threats.
    - **Cons**: Prone to false positives, as legitimate activities that deviate from the baseline may be flagged as threats.
3. **Hybrid Detection**:
    - Combines signature-based and anomaly-based techniques to enhance detection capabilities.
    - Provides a balance between detecting known attacks and identifying novel threats, while minimizing false positives and false negatives.

Key Components of IDS:

1. **Sensors**: Collect data from network traffic or host activity. In NIDS, sensors monitor traffic on network segments, while in HIDS, sensors observe local host activity such as system logs or file integrity.
2. **Analyzers**: Process the data collected by sensors and determine whether it constitutes suspicious or malicious activity. They rely on detection algorithms, whether signature-based or anomaly-based.
3. **Alerting Mechanism**: Once the IDS identifies a potential threat, it generates an alert to notify administrators or security teams. Alerts may include detailed information about the attack, such as the source IP, target, and type of intrusion detected.
4. **Management Console**: A centralized interface for configuring, managing, and monitoring the IDS. The console provides visibility into the status of the system, active threats, and past alerts.

Use Cases for IDS:

- **Monitoring for External Threats**: Detects unauthorized access attempts, denial-of-service attacks, and malware infections from external attackers.
- **Identifying Insider Threats**: Monitors user behavior to detect unusual actions from internal users, such as accessing sensitive data or systems they normally wouldn't use.
- **Compliance**: Many industries require organizations to implement intrusion detection mechanisms as part of regulatory compliance (e.g., PCI DSS, HIPAA).
- **Incident Response**: Alerts generated by the IDS help security teams quickly identify and respond to potential security incidents, improving reaction time.

Strengths of IDS:

1. **Visibility**: Provides detailed insights into network or system activity, helping organizations identify suspicious behavior.
2. **Detection of Known Threats**: Signature-based IDS can accurately detect known attacks.
3. **Proactive Monitoring**: Anomaly-based IDS allows organizations to monitor for novel or unexpected threats.

4. **Forensics**: IDS logs can be valuable for post-attack analysis, helping organizations understand the scope and impact of a breach.

CYBER CRIMES AND CONTROL MEASURES

Cybercrimes are criminal activities that involve the use of computers, networks, or the internet to commit illegal acts, often targeting individuals, organizations, or governments. These crimes range from data theft and financial fraud to hacking and cyberterrorism. As the internet and digital systems become integral to society, cybercrime has grown more sophisticated and damaging, posing significant risks to privacy, security, and economic stability.

Types of Cybercrimes:

1. **Hacking**:
    - Unauthorized access to systems or networks with malicious intent.
    - Hackers exploit vulnerabilities in software or security protocols to gain access to sensitive data or cause damage.
    - **Example**: Website defacement, data breaches, or malware infections.
2. **Phishing**:
    - A form of social engineering where attackers impersonate legitimate entities (e.g., banks, government agencies) to trick users into divulging sensitive information like passwords or credit card numbers.
    - **Example**: Fake emails or websites that prompt victims to input confidential information.
3. **Identity Theft**:
    - Stealing someone's personal information (e.g., social security number, bank details) to commit fraud or other illegal activities.
    - **Example**: Opening credit accounts in someone else's name, filing fraudulent tax returns.
4. **Ransomware**:
    - Malicious software that encrypts the victim's files or system, rendering them inaccessible until a ransom is paid (usually in cryptocurrency).
    - **Example**: The 2017 WannaCry ransomware attack, which affected systems worldwide.
5. **Cyberterrorism**:
    - The use of digital attacks to create fear or disrupt critical infrastructures, such as power grids, transportation systems, or financial institutions.
    - **Example**: Coordinated attacks against government agencies or critical infrastructure, like the 2015 attack on Ukraine's power grid.
6. **Distributed Denial of Service (DDoS) Attacks**:
    - Overwhelming a system, website, or network with excessive traffic from multiple sources, rendering it unusable for legitimate users.
    - **Example**: Attacks on high-profile websites to cause disruption or blackmail businesses.
7. **Financial Fraud and Cyber Extortion**:
    - Crimes involving illegal access to financial information or demanding ransom for sensitive data.
    - **Example**: Online banking fraud, ATM skimming, or demanding ransom in exchange for not leaking sensitive data.
8. **Cyberstalking and Harassment**:
    - Using digital communication to stalk, harass, or threaten individuals.
    - **Example**: Repeatedly sending threatening emails, or using social media to spread false information about a person.
9. **Child Exploitation**:

- Crimes involving the abuse or exploitation of children, often involving the production or distribution of illegal materials.
- **Example**: Child pornography or using the internet to groom minors for sexual exploitation.

10. **Intellectual Property Theft**:
    - Illegal downloading, copying, or distribution of copyrighted materials, including software, movies, music, and proprietary business information.
    - **Example**: Software piracy, movie leaks, or industrial espionage.

Control Measures to Combat Cybercrime:

Effective control measures to prevent and mitigate the impact of cybercrimes involve a combination of technological, legal, and human strategies. Here are some key approaches:

1. Technological Controls:

- **Firewalls and Intrusion Detection/Prevention Systems (IDS/IPS)**: Prevent unauthorized access to networks and detect suspicious activities.
- **Encryption**: Ensures sensitive data is secure in transit and storage by converting it into an unreadable format without the proper decryption key.
- **Anti-malware and Anti-virus Software**: Detects and removes malicious software from systems and networks.
- **Security Patches and Updates**: Regularly updating software and systems to fix known vulnerabilities that attackers could exploit.
- **Multi-factor Authentication (MFA)**: Requires multiple forms of verification (e.g., password and fingerprint) to access sensitive systems, making unauthorized access harder.
- **Network Monitoring and Logging**: Continuous monitoring of network traffic to detect and respond to abnormal activities in real-time.
- **Backup Systems**: Regularly backing up data and systems to ensure recovery in the event of ransomware attacks or data loss.

2. Legal and Regulatory Measures:

- **Cybersecurity Laws**: Countries have implemented various laws and regulations to combat cybercrime, such as:
    - **Computer Fraud and Abuse Act (CFAA)** in the U.S.
    - **General Data Protection Regulation (GDPR)** in the EU, which ensures privacy and data protection.
    - **Information Technology Act, 2000** in India, which criminalizes certain cyber activities.
- **International Cooperation**: Given the global nature of cybercrime, countries collaborate through treaties (e.g., the **Budapest Convention** on Cybercrime) and organizations like **Interpol** to investigate and prosecute cybercriminals across borders.
- **Penalties and Sentencing**: Strict penalties for cybercriminals, including fines, imprisonment, and forfeiture of assets, serve as a deterrent.
- **Data Privacy Regulations**: Laws that require organizations to safeguard personal data, report data breaches, and take accountability for cyber incidents (e.g., GDPR).

3. Organizational Measures:

- **Security Awareness Training**: Educating employees about common cyber threats (e.g., phishing, social engineering) and best practices (e.g., strong password policies) to prevent breaches caused by human error.
- **Incident Response Plans**: Having a clear protocol to follow in case of a cyber incident, including steps for detection, containment, investigation, and recovery.
- **Access Control Policies**: Limiting access to sensitive data and systems only to authorized personnel, using principles like **least privilege** and **role-based access control (RBAC)**.
- **Regular Security Audits**: Periodically reviewing and assessing an organization's cybersecurity practices and defenses to identify gaps and improve policies.
- **Cybersecurity Insurance**: Provides financial coverage for losses due to cyberattacks, including the cost of restoring data, legal fees, and public relations efforts.

4. International and Governmental Efforts:

- **Cybersecurity Agencies**: National cybersecurity agencies (e.g., U.S. **Cybersecurity and Infrastructure Security Agency**, **ENISA** in Europe) that develop policies, provide guidance, and support private and public sectors in preventing cybercrime.
- **CERT (Computer Emergency Response Team)**: National and regional CERTs help organizations respond to cybersecurity incidents by providing technical support and coordination.
- **Public Awareness Campaigns**: Government-led initiatives that raise awareness of cybercrime risks and encourage citizens to adopt safe online behaviors (e.g., using strong passwords, avoiding suspicious links).

5. Cybercrime Investigation and Forensics:

- **Digital Forensics**: Involves collecting, preserving, and analyzing digital evidence to investigate cybercrimes, such as recovering deleted files or tracing the source of an attack.
- **Threat Intelligence Sharing**: Organizations and governments often collaborate to share intelligence on emerging cyber threats, malware signatures, and attack trends to enhance collective defenses.

6. Penetration Testing:

- **Ethical Hacking**: Organizations hire ethical hackers or conduct penetration tests to simulate cyberattacks on their systems, helping to uncover vulnerabilities before they are exploited by criminals.