Chapter 1

Overview of Computer security

Table of Contents

Computer Security Concepts:

Computer security is about protecting information systems from unauthorized access, ensuring that the data remains correct and accessible when needed.

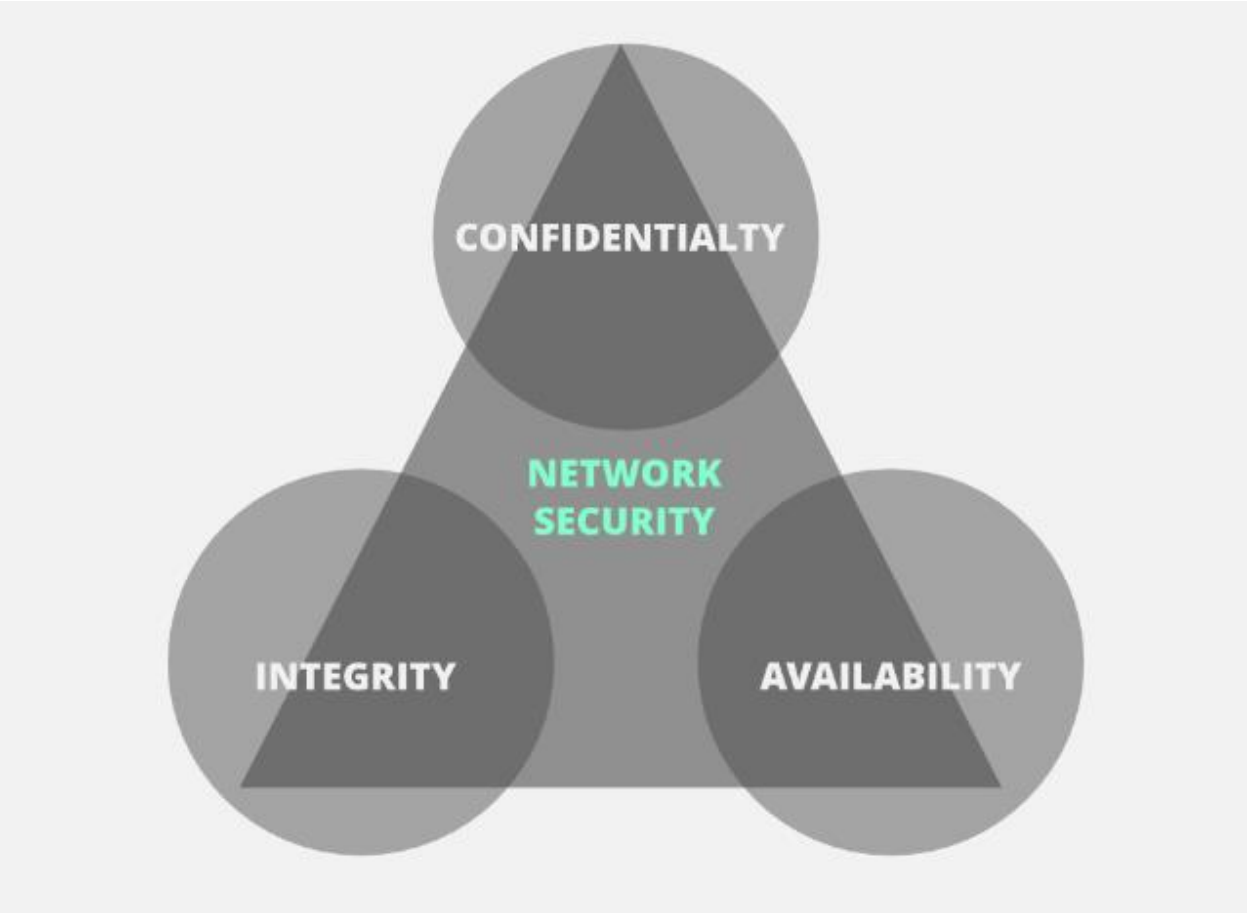
The National Institute of Standards and Technology (NIST) defines computer security as the protection given to an automated information system to achieve the goals of maintaining the integrity, availability, and confidentiality of its resources.

These resources include hardware, software, firmware, data, and telecommunications.

When talking about network security, the **CIA** triad is one of the most important models or objectives which is designed to guide policies for information security within an organization.

CIA stands for :

1. Confidentiality
2. Integrity
3. Availability



1. **Confidentiality:**
    - **Data Confidentiality:** Ensures that private or sensitive information is not accessed or disclosed to unauthorized individuals.

- **Privacy:** Ensures that individuals have control over what personal information is collected, stored, and shared, and who has access to that information.

2. **Integrity:**
   - **Data Integrity:** Ensures that information and programs can only be modified in specified and authorized ways.
   - **System Integrity:** Ensures that a system functions correctly and reliably, without unauthorized manipulation, whether accidental or deliberate.
3. **Availability:**
   - Ensures that systems operate efficiently and that services are accessible to authorized users without interruptions.

Computer Security Challenges:

1. **Complexity and Cost:**
    - Ensuring security is not straightforward. It demands extensive research and substantial financial investment.
2. **Anticipating Attacks:**
    - Security features must be designed with potential attacks in mind, making it a proactive rather than reactive process.
3. **Strategic Deployment:**
    - Determining where and how to implement various security mechanisms is crucial and requires careful planning.
4. **Constant Monitoring:**
    - Security is not a one-time setup; it needs ongoing surveillance to detect and respond to new threats.
5. **Comprehensive Mechanisms:**
    - Effective security involves more than just a single algorithm or protocol. It requires a multifaceted approach combining various tools and techniques.

Security Concept and Relationships:

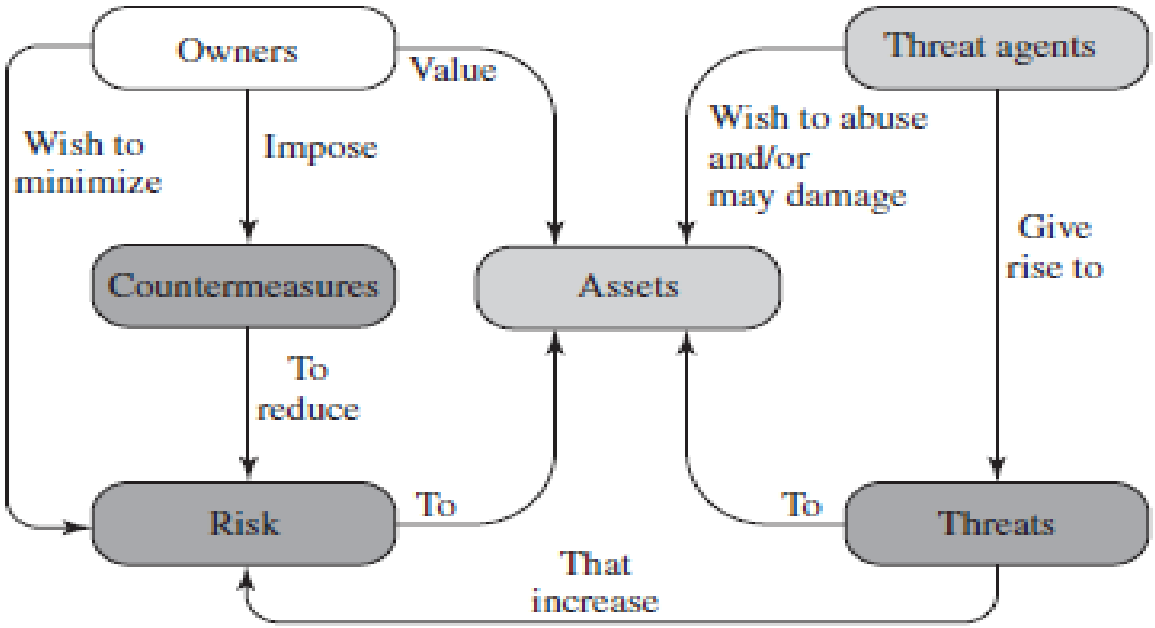


Figure 1.1 **Security Concepts and Relationships**

The diagram in Figure 1.1 illustrates the key concepts and relationships in computer security. Here's a step-by-step explanation of how these elements interact:

1. **Owners:**
    - **Role:** Owners are the stakeholders who possess or are responsible for the assets.
    - **Actions:** They wish to minimize risks associated with their assets and impose countermeasures to protect them.

2. **Assets:**
    - **Description:** Assets are valuable resources, including data, hardware, software, and systems that need protection.
    - **Value:** Assets hold value to the owners and are essential for their operations or goals.
3. **Threat Agents:**
    - **Role:** Threat agents are entities (individuals, groups, or environmental factors) that pose potential harm to the assets.
    - **Actions:** They wish to abuse, exploit, or damage the assets.
4. **Threats:**
    - **Description:** Threats are potential events or actions that can cause harm to assets. They arise from threat agents.
    - **Impact:** Threats increase the level of risk to the assets.
5. **Risk:**
    - **Description:** Risk is the potential for loss or damage when a threat exploits a vulnerability in an asset.
    - **Management:** Owners wish to minimize risk to protect their assets.

6. **Countermeasures:**
    - **Description:** Countermeasures are actions, devices, procedures, or techniques that reduce risk by protecting assets against threats.
    - **Purpose:** They are imposed by owners to mitigate the impact of threats and minimize risk.

Relationships:

- **Owners and Assets:** Owners value their assets and impose countermeasures to protect them.
- **Threat Agents and Assets:** Threat agents wish to exploit or damage the assets, creating threats.
- **Threats and Risk:** Threats, posed by threat agents, increase the risk to assets.
- **Countermeasures and Risk:** Countermeasures, imposed by owners, are aimed at reducing the risk to assets by addressing threats.

Computer Security:

- **Definition:** Focuses on the protection of standalone computers and the data and processes they handle.
- **Scope:** Involves safeguarding the integrity, availability, and confidentiality of the system and its components.
- **Examples:**
  - Protecting against malware (viruses, worms, trojans).
  - Implementing firewalls and antivirus software.
  - Ensuring secure access controls and user authentication.

Information Security:

- **Definition:** Encompasses the protection of all forms of information, whether in digital, physical, or other formats, from unauthorized access, use, disclosure, disruption, modification, or destruction.
- **Scope:** Broader than computer security, it includes data integrity, confidentiality, and availability regardless of the medium.
- **Examples:**
  - Encrypting sensitive data.
  - Implementing policies for information handling and disposal.

Network Security:

- **Definition:** Focuses on protecting the integrity, availability, and confidentiality of information as it is transmitted across or accessed via networks.
- **Scope:** Involves securing the infrastructure and communication channels of both private and public networks.
- **Examples:**
    - Implementing secure communication protocols (e.g., SSL/TLS).
    - Using intrusion detection and prevention systems (IDS/IPS).
    - Ensuring robust firewall configurations and network segmentation.

Differences:

- **Computer Security:** Primarily concerned with individual machines and their software and hardware components.
- **Information Security:** Broad focus on all information, regardless of its format or location.
- **Network Security:** Concentrates on protecting data during transmission and securing network infrastructure.

Threats, Attacks, and Assets:

1. Threats:

- **Definition:** Potential events or actions that could cause harm to assets. Threats can be intentional or accidental, and can come from various sources, such as individuals, groups, or environmental factors.
- **Types of Threats:**
  - **Human Threats:** Malicious insiders, hackers, social engineers.
  - **Technical Threats:** Malware, software vulnerabilities, system failures.
  - **Physical Threats:** Theft, natural disasters, power outages.
  - **Environmental Threats:** Fire, flood, earthquakes.

2. Attacks:

- **Definition:** Deliberate actions by someone trying to take advantage of weaknesses in systems or information to cause harm or gain unauthorized access.
- **Types of Attacks:**
  - **Passive Attacks:** Eavesdropping on or monitoring of communications, such as traffic analysis and packet sniffing.

- **Active Attacks:** Attempts to alter or destroy data, disrupt services, or gain unauthorized access. Examples include:
    - **Phishing:** Deceptive attempts to obtain sensitive information by masquerading as a trustworthy entity.
    - **Denial-of-Service (DoS):** Overwhelming a system to make it unavailable to legitimate users.
    - **Man-in-the-Middle (MitM):** Intercepting and possibly altering communications between two parties without their knowledge.
    - **SQL Injection:** Inserting malicious SQL code into a database query to manipulate the database.

3. Assets:

- **Definition:** Valuable resources that need to be protected. These can include information, hardware, software, systems, and networks.
- **Types of Assets:**
    - **Information Assets:** Data, intellectual property, customer information, financial records.
    - **Hardware Assets:** Computers, servers, network devices, mobile devices.
    - **Software Assets:** Applications, operating systems, utilities, firmware.

- **System Assets:** Databases, information systems, communication systems.
- **Network Assets:** Network infrastructure, routers, switches, communication channels.

	Availability	Confidentiality	Integrity
Hardware	Equipment is stolen or disabled, thus denying service.	An unencrypted CD-ROM or DVD is stolen.	
Software	Programs are deleted, denying access to users.	An unauthorized copy of software is made.	A working program is modified, either to cause it to fail during execution or to cause it to do some unintended task.
Data	Files are deleted, denying access to users.	An unauthorized read of data is performed. An analysis of statistical data reveals underlying data.	Existing files are modified or new files are fabricated.
Communication Lines and Networks	Messages are destroyed or deleted. Communication lines or networks are rendered unavailable.	Messages are read. The traffic pattern of messages is observed.	Messages are modified, delayed, reordered, or duplicated. False messages are fabricated.

Figure: Computer and Network Assets, with Examples of Threats

Security Requirements:

Security requirements are critical measures and controls that organizations implement to protect their information systems and data from various threats.

1. **Access Control:**
    - **What it means:** Only let authorized users and devices access the information system, and control what actions they can perform.
    - **Example:** Using passwords and user permissions to ensure only certain employees can view or edit sensitive files.
2. **Awareness and Training:**
    - **What it means:** Educate managers and users about security risks and the rules they need to follow, and train them on how to handle security-related tasks.
    - **Example:** Conducting regular security training sessions and informing employees about phishing scams.
3. **Audit and Accountability:**
    - **What it means:** Keep records of system activities to monitor and investigate any unauthorized actions, and ensure users can be held responsible for their actions.
    - **Example:** Logging user activities and reviewing these logs to detect suspicious behavior.

4. **Certification, Accreditation, and Security Assessments:**
    - **What it means:** Regularly check the security measures in place to ensure they work effectively, fix any issues found, approve the system for use, and continuously monitor security.
    - **Example:** Performing annual security audits and vulnerability assessments.
5. **Configuration Management:**
    - **What it means:** Maintain and update the system's baseline settings and inventory, and enforce security settings throughout the system's lifecycle.
    - **Example:** Keeping an inventory of all hardware and software, and ensuring all systems are configured securely.
6. **Contingency Planning:**
    - **What it means:** Prepare plans for emergency responses, backups, and disaster recovery to ensure critical information and operations continue during emergencies.
    - **Example:** Having a disaster recovery plan that includes data backups and alternate work sites.

7. **Identification and Authentication:**
    - **What it means:** Verify the identities of users, processes, and devices before allowing them to access the system.
    - **Example:** Implementing multi-factor authentication for system access.
8. **Incident Response:**
    - **What it means:** Develop a capability to handle security incidents, including preparation, detection, analysis, containment, recovery, and reporting.
    - **Example:** Having an incident response team and plan to quickly address and recover from security breaches.
9. **Maintenance:**
    - **What it means:** Regularly perform maintenance on information systems and control the tools and personnel involved in maintenance activities.
    - **Example:** Scheduling regular system updates and patches to keep software secure.

Security Design Principles:

Security design principles provide a framework for creating systems that are secure by design.

By applying these principles, organizations can build systems that are more resilient to attacks, easier to manage, and less prone to security breaches.

Implementing these principles helps ensure that security is integrated into the system from the ground up, rather than being an afterthought.

some key security design principles are:

1. **Least Privilege:**
    - Give users and systems the minimum level of access necessary to perform their tasks.
    - **Example:** An employee in accounting should not have access to the company's marketing data.
2. **Separation of Duties:**
    - Divide critical tasks among different people to reduce the risk of fraud or error.
    - **Example:** One person approves expense reports, and another person processes the payments.

3. **Defense in Depth:**
    - Use multiple layers of security controls to protect resources, so if one layer fails, others still provide protection.
    - **Example:** Using firewalls, intrusion detection systems, and encryption together to protect data.
4. **Fail-Safe Defaults:**
    - Systems should default to a secure state in the event of a failure.
    - **Example:** If a system error occurs, access to sensitive data should be denied rather than granted.
5. **Economy of Mechanism:**
    - Keep designs as simple and small as possible to reduce complexity and potential vulnerabilities.
    - **Example:** Using straightforward, well-tested algorithms rather than complex, obscure ones.
6. **Complete Mediation:**
    - Every access to a resource must be checked for authorization.
    - **Example:** Every time a user tries to access a file, the system checks their permissions, not just the first time.

7. **Open Design:**
    - Security should not depend on the secrecy of the design or implementation.
    - **Example:** Using publicly vetted encryption algorithms rather than proprietary, secret ones.
8. **Separation of Privilege:**
    - Require multiple conditions to grant access to sensitive operations.
    - **Example:** Requiring two separate keys to launch a nuclear missile.
9. **Psychological Acceptability:**
    - Security mechanisms should not make the system too difficult to use.
    - **Example:** Implementing user-friendly multi-factor authentication that does not frustrate users.
10. **Modularity:**
    - Design the system in modular components, so each component can be independently secured and updated.
    - **Example:** Separating a web application into different modules such as authentication, database access, and user interface.

11. **Fail Securely:**
    - Ensure that when systems fail, they do so in a secure manner.
    - **Example:** A bank's ATM machine locking up and denying further transactions if it detects tampering.
12. **Minimize Attack Surface:**
    - Reduce the number of ways an attacker can exploit a system.
    - **Example:** Disabling unused services and ports on a server.

Attack Surfaces and Attack Trees:

Attack surfaces refer to all the points in a system where an attacker could try to exploit vulnerabilities.

These are the parts of a system that can be reached and attacked.

Examples of Attack Surfaces

1. **Open Ports:**
    - These are points where the system connects to the internet, like a door to your house. If these are left unprotected, hackers can enter.
    - **Example:** A website's server with open ports for web traffic.

2. **Services Inside a Firewall:**
    - These are internal services that should be protected but might still have vulnerabilities.
    - **Example:** A company's internal email server.
3. **Data Processing Code:**
    - Any code that handles incoming data, such as emails or documents, can be exploited if not secure.
    - **Example:** Software that processes attachments in emails.
4. **Interfaces and Forms:**
    - Web forms, SQL databases, and other user inputs are points where attackers can try to insert harmful data.
    - **Example:** A login form on a website.

Types of Attack Surfaces

1. **Network Attack Surface:**
    - Vulnerabilities in the way a system connects over networks like the internet.
    - **Example:** Network protocol flaws that allow hackers to disrupt communication.

2. **Software Attack Surface:**
    - Vulnerabilities in the software, including applications, utilities, and operating systems.
    - **Example:** Bugs in a web server's code that can be exploited by hackers.
3. **Human Attack Surface:**
    - Vulnerabilities caused by people, including social engineering, mistakes, and insiders with malicious intent.
    - **Example:** A worker accidentally clicking on a malicious link.

Why Attack Surface Analysis is Important

- **Purpose:** To identify all the vulnerable points in a system.
- **Benefits:**
    - **Awareness:** Helps developers and security teams understand where security needs to be improved.
    - **Reduction:** Identifies ways to make the attack surface smaller, making it harder for attackers.
    - **Prioritization:** Guides where to focus testing and security enhancements.
    - **Improvement:** Suggests changes to services or applications to enhance security.

Attack Tree:

An attack tree is a visual tool used to understand how a security breach can occur.

It breaks down the process of an attack into smaller, manageable parts, showing different ways an attacker might achieve their goal.

Key Components of an Attack Tree:

1. **Root Node:**
    - This is the main goal of the attacker. It's what they ultimately want to achieve.
    - **Example:** In an internet banking scenario, the root node might be "Compromise a user's account."
2. **Branches and Subnodes:**
    - These are steps or subgoals that lead towards the main goal. Each step might be broken down into further substeps.
    - **Example:** To compromise a user's account, an attacker might need to steal login credentials or bypass security measures.
3. **Leaf Nodes:**
    - These are the final, detailed steps that directly initiate an attack. They represent specific actions an attacker can take.

- **Example:** Stealing a user's password through phishing.

4. **AND-Nodes and OR-Nodes:**
    - **AND-Nodes:** To achieve this goal, all subgoals under this node must be met.
    - **OR-Nodes:** To achieve this goal, at least one of the subgoals under this node must be met.
    - **Example:** If a node is an AND-node, an attacker might need both the user's password and physical access to their device. If it's an OR-node, having just the password might be enough.

Example: Internet Banking Authentication Attack Tree

Let's consider an example attack tree for compromising an internet banking account.

- **Root Node:** Compromise a user's account.
- **Three Main Areas of Attack:**
    1. **User Terminal and User (UT/U):**
        - Attacks targeting the user's equipment and actions.
        - **Example:** Stealing a smartcard, phishing for passwords.
    2. **Communications Channel (CC):**
        - Attacks focusing on the communication links.

- **Example:** Intercepting data transmitted over the internet.

3. **Internet Banking Server (IBS):**
    - Attacks against the servers hosting the internet banking application.
    - **Example:** Exploiting server vulnerabilities to access account data.

Five Overall Attack Strategies:

1. **Steal User's Credentials (UT/U):** Phishing attacks to get passwords.
2. **Exploit Weak Communication (CC):** Man-in-the-middle attacks to intercept data.
3. **Compromise User's Device (UT/U):** Installing malware on the user's device.
4. **Exploit Server Vulnerabilities (IBS):** Attacking the bank's server directly.
5. **Social Engineering (UT/U):** Tricking the user into revealing their information.

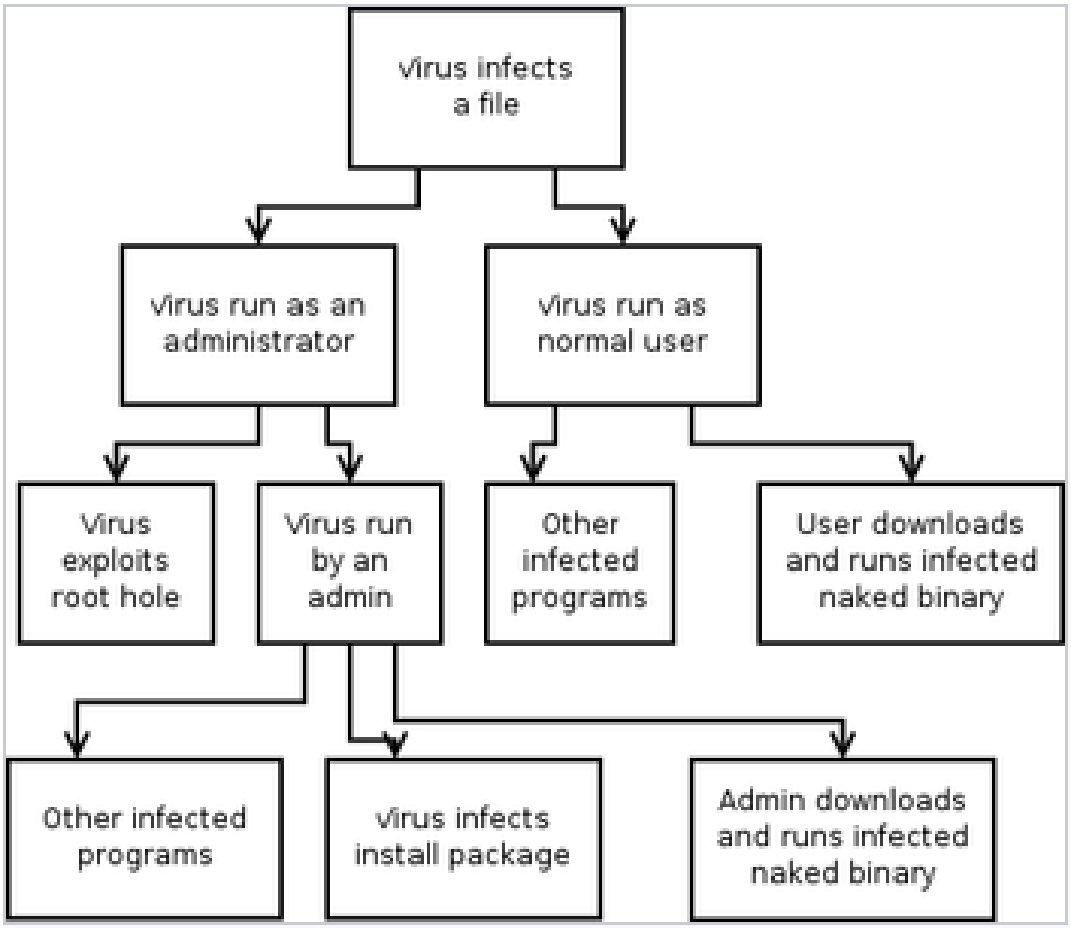


Figure: Attack tree for computer viruses

Computer Security Strategy:

The overall strategy for providing computer security typically involves three main aspects: specification/policy, implementation/mechanisms, and correctness/assurance.

Security Policy:

- **Specification/Policy:** This aspect defines what the security scheme is supposed to achieve.

  It involves formal or informal statements of rules and practices that regulate how a system or organization provides security services to protect sensitive and critical resources.

  Factors to consider include the value of assets being protected, system vulnerabilities, potential threats, and trade-offs between ease of use and security, as well as the cost of security versus the cost of failure and recovery.

Security Implementation:

- **Prevention:** This involves setting up security measures to prevent attacks.

  Examples include using secure encryption algorithms and access control mechanisms to protect data confidentiality.

- **Detection:** While absolute protection may not always be feasible, detecting security attacks is practical in many cases. Intrusion detection systems and other monitoring tools can help identify unauthorized access or denial of service attacks.

- **Response:** If an attack is detected, the system should be able to respond effectively to halt the attack and prevent further damage.
- **Recovery:** This involves measures to recover from security breaches, such as using backup systems to restore data integrity after a compromise.

Assurance and Evaluation:

- **Assurance:** Assurance refers to the degree of confidence one has that the security measures, both technical and operational, work as intended to protect the system and the information it processes.

  While absolute proof of correctness may not be feasible, assurance is expressed as a degree of confidence based on system design and implementation.
- **Evaluation:** Evaluation involves examining a computer product or system with respect to certain criteria, which may include testing and formal analytic or mathematical techniques.

  The goal is to develop evaluation criteria that can be applied to any security system for making product comparisons.