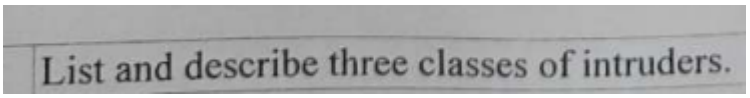
Perimeter Security

Introduction to Intruders

Intrusion into a system is a critical issue in network and system security. It refers to unauthorized access, often for malicious purposes. Intruders exploit weaknesses in systems to gain access, escalate privileges, and hide their activities. Understanding intrusion techniques and detection methods helps in safeguarding systems effectively.

---

1. Types of Intruders


List and describe three classes of intruders.

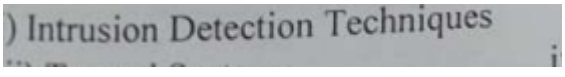
Intruders can be classified into three types based on their behavior and access methods:

1. **Masquerader**: An outsider pretending to be an authorized user.
    a. Example: A hacker using stolen credentials to access a system.
2. **Misfeasor**: A legitimate user abusing their access rights.
    a. Example: An employee misusing sensitive company data.
3. **Clandestine User**: A person who gains access secretly and avoids detection.
    a. Example: A spy who hides their activities while accessing classified data.

These categories help in understanding the methods intruders use, guiding detection strategies.

---

2. Intrusion Techniques


) Intrusion Detection Techniques

Intruders follow systematic steps to breach a system:

1. **Target Acquisition and Information Gathering**:
    - Collect data about the system using public sources or scanning tools.
2. **Initial Access**:
    - Exploit vulnerabilities like weak passwords or outdated software to enter the system.
3. **Privilege Escalation**:
    - Gain higher-level access to perform restricted actions, such as modifying files or disabling security features.
4. **Covering Tracks**:
    - Delete logs or use tools to hide unauthorized activities, ensuring they remain undetected.

---

3. Password Attacks

Passwords are a primary target for intruders. Attacks include:

- **Password Guessing**:
    - Methods: Guessing based on default passwords, user information (birthdays, names), or exhaustive searches.
    - Example: Trying “password123” for a known username.
    - Prevention: Use strong, unpredictable passwords.
- **Password Capture**:
    - Techniques: Watching users (shoulder surfing), trojans, insecure network protocols (e.g., telnet), or cached data.
    - Example: A trojan that logs keystrokes.
    - Prevention: Educating users to avoid insecure logins and use secure methods like HTTPS.

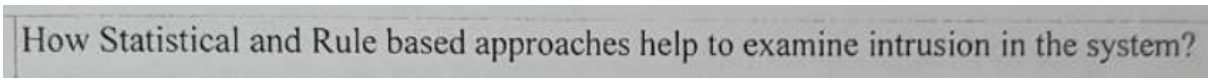
---

4. Intrusion Detection Systems (IDS)

Since preventing all intrusions is impossible, **Intrusion Detection Systems** aim to detect and respond to suspicious activities:

- **Purpose**:
  - Detecting security breaches.
  - Block intrusions quickly or collect data for future improvements.
- **Assumption**: Intruders exhibit behavior different from legitimate users.

Detection Approaches:


How Statistical and Rule based approaches help to examine intrusion in the system?

1. **Statistical Anomaly Detection**:
   - Monitors user/system behavior and flags significant deviations.
   - Examples:
     - **Threshold Detection**: If login attempts exceed a set number, flag it.
     - **Profile-Based Detection**: Identifies abnormal patterns in system usage based on historical data.
   - Limitation: May generate false positives if legitimate users behave unexpectedly.
2. **Rule-Based Detection**:
   - Uses predefined rules to identify intrusions.
   - Types:
     - **Anomaly Detection**: Matches current behavior against rules generated from historical patterns.
     - **Penetration Identification**: Uses expert-defined rules to detect known vulnerabilities.

---

5. Audit Records

Audit logs serve as the foundation for IDS:

- **Native Records**: Built-in OS logs (e.g., login attempts).
- **Detection-Specific Records**: Custom logs designed to capture specific security data.
  - Example: Logging every file accessed during a session.

    - Trade-off: Custom logs add system overhead but enhance detection capabilities.

---

6. Challenges in Intrusion Detection

- **Base-Rate Fallacy**:
    - IDS needs a balance:
        - Detect a high percentage of intrusions.
        - Minimize false alarms.
    - Too many false positives: Alerts are ignored.
    - Too few detections: Creates a false sense of security.
    - Practical Example: An IDS flagging normal admin tasks as suspicious will waste time.

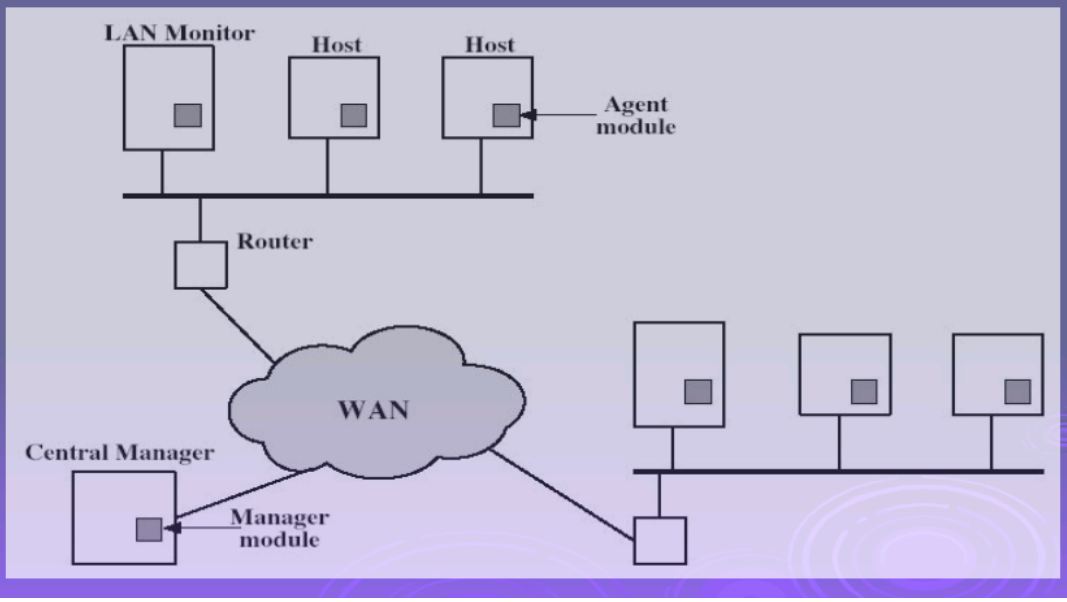
---

7. Distributed Intrusion Detection

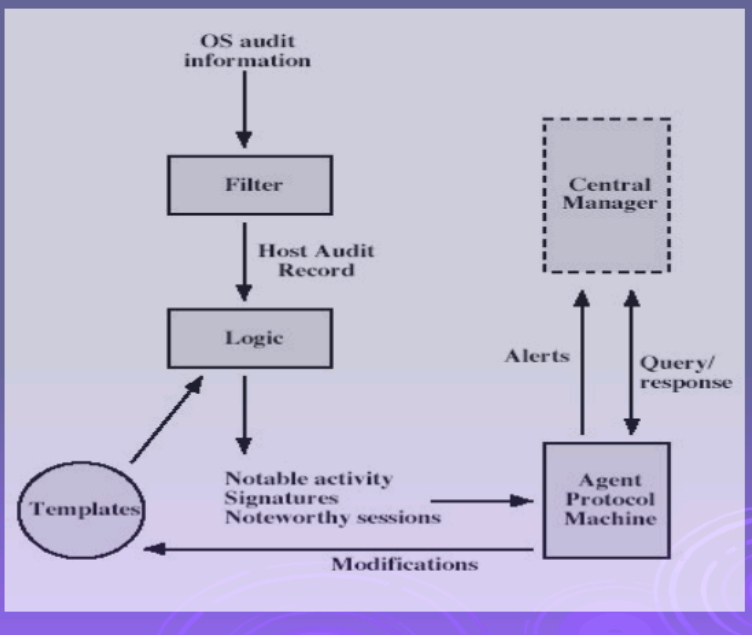
Modern systems are interconnected, requiring coordinated defense:

- **Purpose**: Systems across a network work together to detect and respond to threats.
- **Challenges**:
    - Varying formats of logs/audit records.
    - Ensuring the security of data shared between systems.
    - Choosing a centralized (single control point) or decentralized (peer-to-peer) architecture.

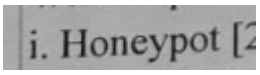
- ## Distributed Intrusion Detection - Architecture



- ## Distributed Intrusion Detection – Agent Implementation



Honeypots



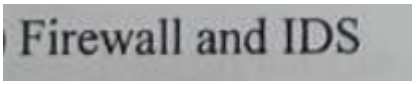
1. **Definition**: Decoy systems designed to attract attackers and divert them from critical systems.
2. **Purpose**:
    - Lure attackers away from sensitive systems.
    - Collect information on attacker behavior.
    - Keep attackers engaged to allow administrators to respond.
3. **Features**:
    - Filled with fabricated data to appear valuable.
    - Instrumented to record detailed attacker activities.
    - Can be standalone systems or a network of interconnected honeypots.
4. **Standards**: Follow guidelines such as those from the IETF Intrusion Detection Working Group.

Password Management

1. **Definition**: Processes and techniques to ensure secure password usage and storage.
2. **How Passwords Work**:
    - **Login**: Identifies user privileges.
    - **Password**: Authenticates the user.
    - Passwords are typically encrypted using cryptographic methods like salted DES or hash functions.
3. **Challenges**:
    - Studies (Purdue 1992, Klein 1990) show users often choose weak, guessable passwords.
4. **Management Techniques**:
    - **Education**:
        - Teach users to create strong passwords (e.g., minimum length, character variety).
        - Many users still ignore guidelines.
    - **Computer-Generated Passwords**:
        - Generate random or pronounceable passwords.

        - May lead to insecure practices like writing passwords down.
    - **Reactive Checking**:
        - Use tools to identify weak passwords after creation.
        - Resource-intensive and leaves passwords vulnerable until detected.
    - **Proactive Checking**:
        - Verify password strength during creation.
        - Methods include:
            - Rule enforcement (length, complexity).
            - Dictionary checks.
            - Advanced algorithms like Markov models or Bloom filters

What is a Firewall?

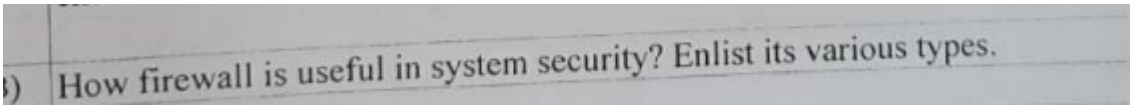


- Acts as a **choke point** between trusted and untrusted networks.
- Controls and monitors traffic based on security rules.
- Features:
    - **NAT (Network Address Translation)**: Masks internal IPs.
    - **VPN (Virtual Private Network)**: Uses IPSec for secure connections.
    - **Traffic Monitoring**: Detects and logs abnormal behavior.

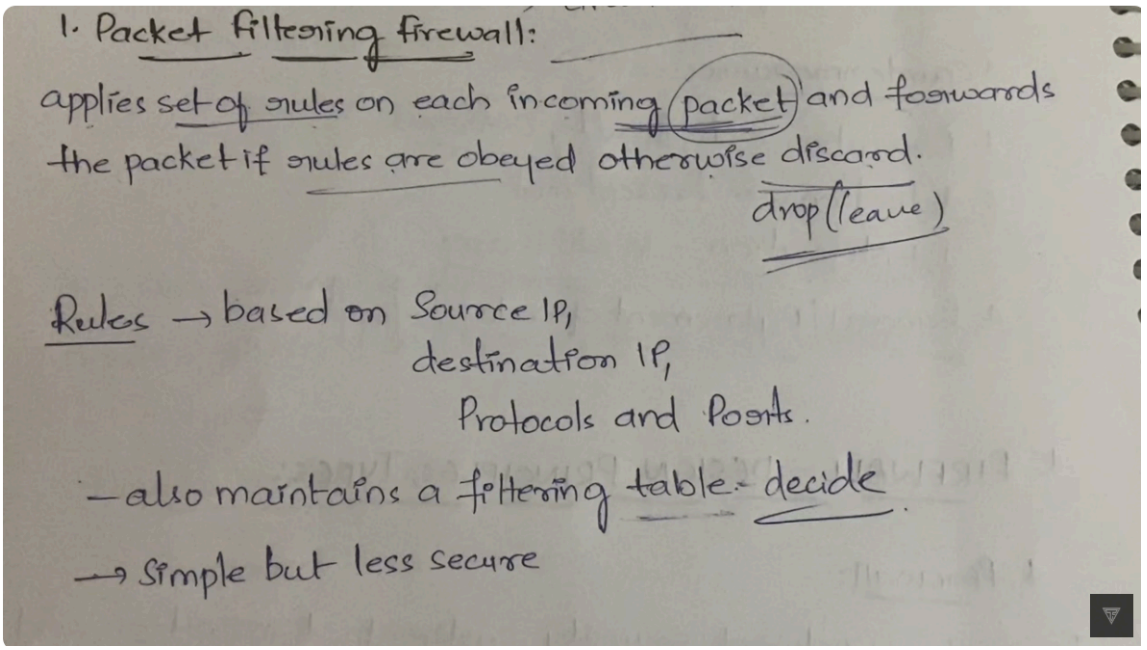
Limitations

- Cannot stop:
    - Attacks bypassing the firewall (e.g., through USB drives or trusted services like SSH/SSL).
    - Internal threats from employees or trusted users.
    - Transfer of virus-infected files.

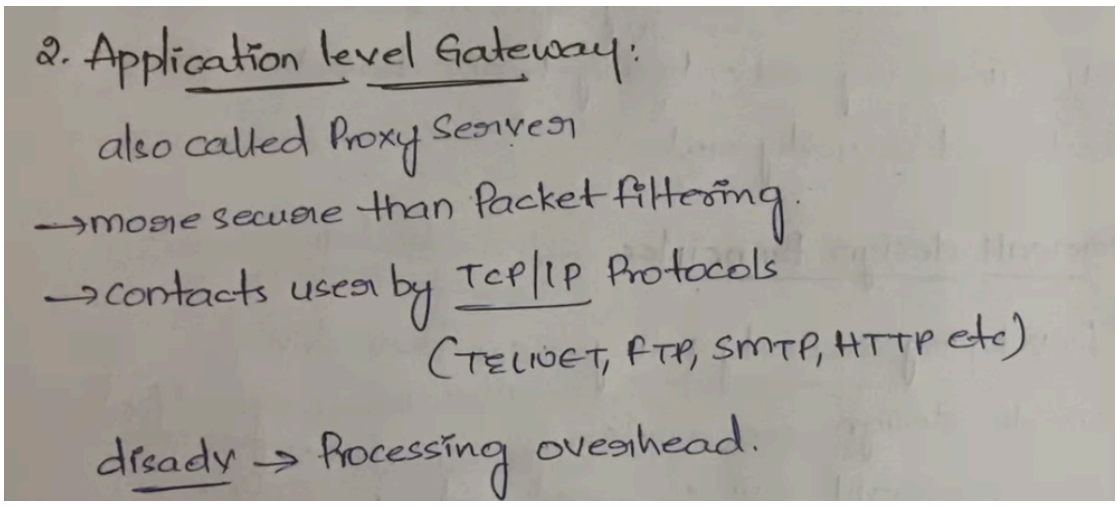
Types of Firewalls

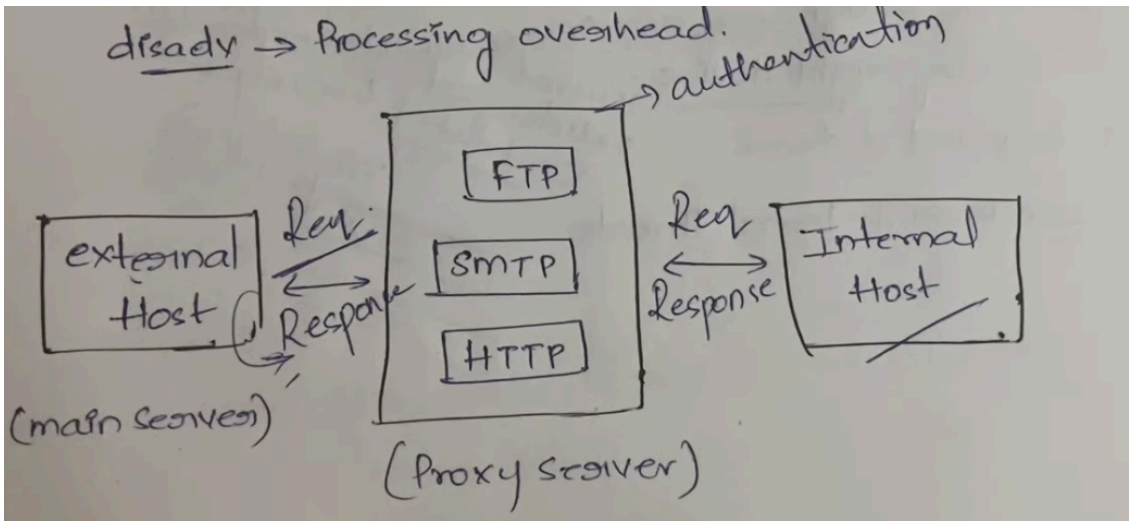


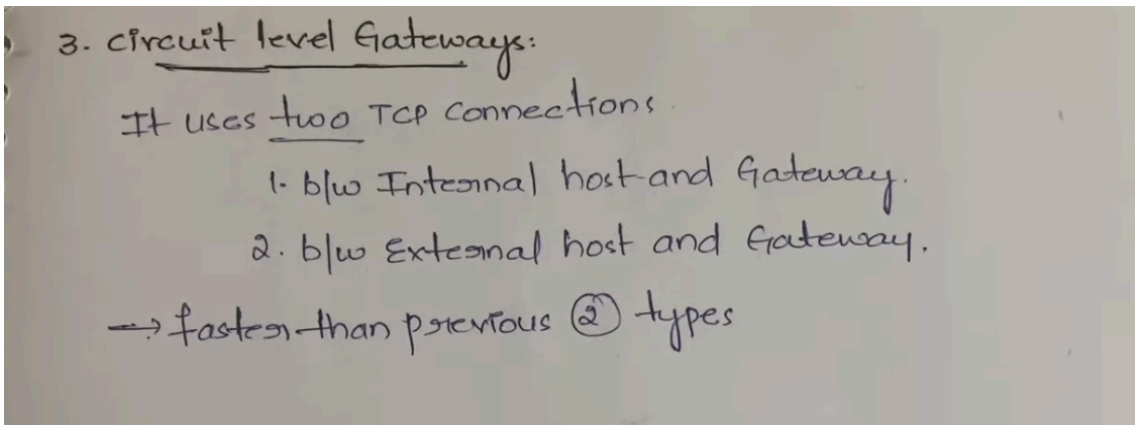
Focus on Firewall design configurations in terms of implementation.

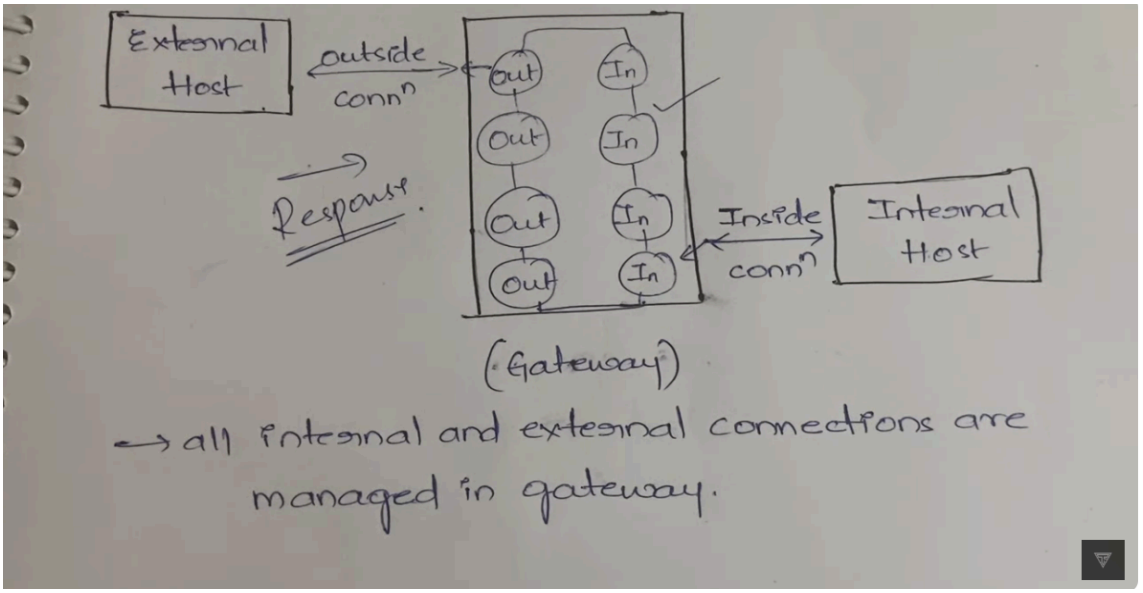
1. **Packet Filters**:
   - 

   - Examine individual packets without context.
   - Default Policies:
     - Deny all except explicitly permitted.
     - Allow all except explicitly denied.
   - **Attacks**:
     - **IP Spoofing**: Fake source IPs.
       Counter: Add filters.
     - **Source Routing**: Bypass default routes.
       Counter: Block such packets.
     - **Tiny Fragments**: Split packet headers to bypass rules.
       Counter: Reassemble or discard them.
2. **Stateful Packet Filters**:
   - Track and validate client-server sessions.
   - Better than packet filters at detecting invalid packets.
3. **Application-Level Gateways (Proxies)**:

- 

- 

  - Proxies for specific services (e.g., web or email).
  - Validates and processes requests securely.
  - Logs and audits traffic.
4. **Circuit-Level Gateways**:
   - 



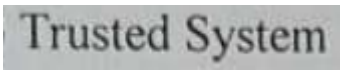
- Relays TCP connections securely.
- Doesn’t inspect data contents.
- Commonly uses the SOCKS protocol.

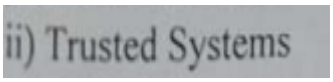
5. **Bastion Hosts**:
   - Highly secure systems designed for hostile environments.
   - Features:
     - Hardened OS.
     - Limited services and extra authentication.
     - Proxies are independent and minimal.

Firewall Configurations

- **Single Firewall**: Basic protection.
- **DMZ (Demilitarized Zone)**:
  - Isolates public-facing services from internal systems.
  - Uses multiple firewalls for enhanced security.

Trusted Systems





What are Trusted Systems?

- Systems that ensure **data security** based on sensitivity levels.
- Implemented to handle information of varying sensitivity (e.g., confidential, secret).

Multilevel Security (MLS)

- **Subjects (Users)**: Have maximum and current security levels.
- **Objects (Data)**: Have fixed security classifications.

Bell-LaPadula (BLP) Model

1. **No Read Up**:
    - A user cannot read data classified at a higher level than their security clearance.
2. **No Write Down**:
    - A user cannot write to a lower security level, preventing data leakage.

Access Control

- Uses an **Access Control Matrix**:
    - **Subjects**: Active users or processes.
    - **Objects**: Files or resources.
    - **Access Rights**: Define actions (read, write, etc.).

Evaluation Standards

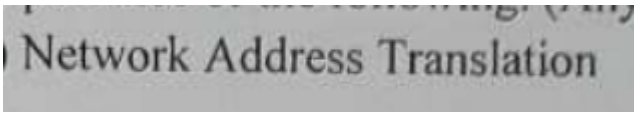
- **Common Criteria**:
    - International standard to evaluate IT security.
    - Two requirements:
        - **Functional**: Crypto, user authentication, data protection, etc.

- **Assurance**: System development, testing, and vulnerability assessment.

Reference Monitor

- Ensures all access requests adhere to security policies.
- Evaluated under frameworks like TCSEC or Common Criteria.

PYQ:



Short Note on NAT (Network Address Translation)

Definition:
Network Address Translation (NAT) is a method used to map multiple private IP addresses to a single public IP address or a pool of public IPs.

Purpose:

- Conserves public IP addresses.
- Provides security by hiding internal IPs from external networks.

Types of NAT:

1. **Static NAT**: Maps one private IP to one public IP.
2. **Dynamic NAT**: Maps private IPs to a pool of public IPs.
3. **PAT (Port Address Translation)**: Maps multiple private IPs to a single public IP using different port numbers (also called Overloading).

Advantages:

- Saves IP address space.
- Adds a layer of security by masking internal addresses.
- Facilitates private network communication with external networks.

Limitations:

- Can cause delays in connection.
- Does not support some protocols like IPsec without configuration.