Dr. Ammar Haider
Assistant Professor
School of Computing

CS3002 Information Security

Intrusion Detection Systems

Reference: Stallings SPP chap 8

Intrusion

- Attempt to break into or misuse an information system
- Intruders may be from outside the network or from legitimate users of the network
- Every intrusion event is an ‘incident’

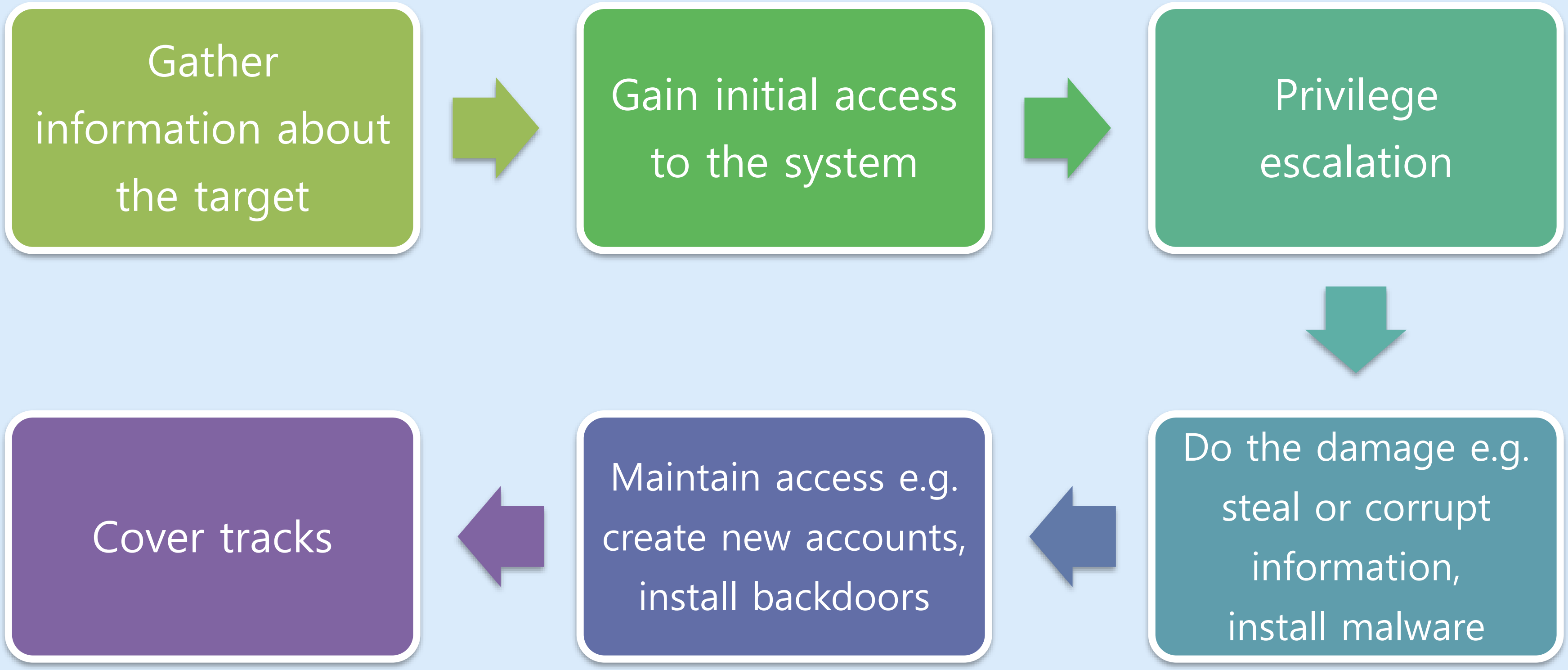
Types of Intruders

- Masquerader: an individual (usually outsider) who is not authorized to use the computer and who penetrates a system's access controls to exploit a legitimate user's account.
- Misfeasor: A legitimate user (usually insider) who access data, program, or resources for which such access is not authorized, or who is authorized for such access but misuses them.
- Clandestine user: An individual who seizes supervisory control of the system and uses this control to evade auditing and access controls or to suppress audit collection. (Can be either inside or outside)

Examples of Intrusion Attacks

- Performing a remote root compromise of an e-mail server
- Defacing a web server
- Guessing and cracking passwords
- Copying a database containing credit card numbers
- Viewing sensitive data (e.g. payroll records and media) without authorization
- Running a packet sniffer on a workstation to capture usernames and passwords
- Using an unattended, logged-in workstation without permission

Intruder’s workflow



Gather information about the target

Gain initial access to the system

Privilege escalation

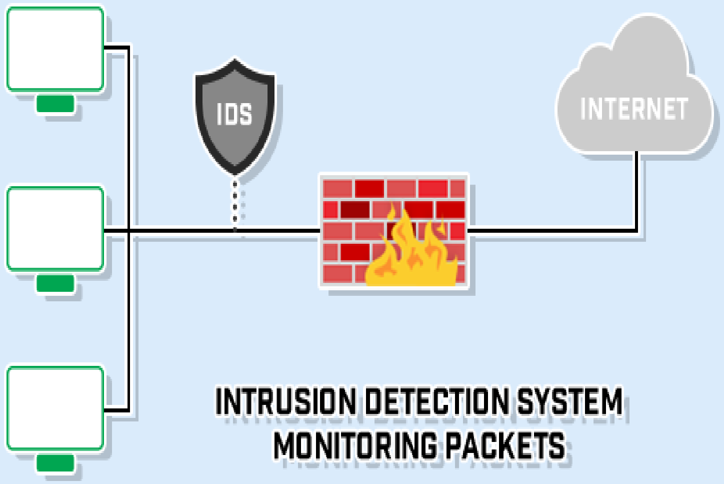
Do the damage e.g. steal or corrupt information, install malware

Maintain access e.g. create new accounts, install backdoors

Cover tracks

Intrusion Detection System (IDS)

- A security service that monitors system events and analyzes them for the purpose of finding, and providing real-time or near real time warning of intrusion attempts.
- Intrusion Detection Systems look for attack signatures (patterns that usually indicate malicious or suspicious intent)



IDS Components

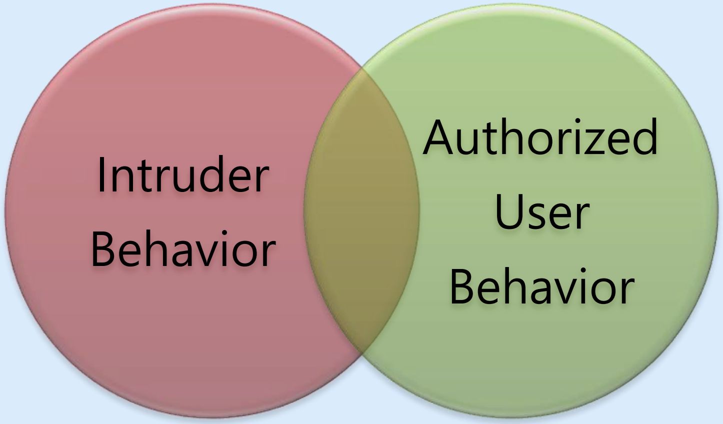
- **Sensors**: responsible for collecting data (network packets, log files, and system call traces)
- **Analyzers**: receive inputs from one or more sensors or from other analyzers. The analyzer is responsible for determining if an intrusion has occurred.
- **User Interface**: enables a user to view output from the system or control behavior of the system.

IDS – Basic Principles

1. If an intruder is detected quickly enough, the intruder can be identified and ejected from the system before any damage. Even if the detection is not that quick, the sooner the intrusion is detected, the less the amount of damage and more quickly the recovery can be achieved.
2. An effective IDS can serve as a deterrent, thus acting to prevent intrusion.
3. Intrusion detection enables the collection of information about intrusion techniques that can be used to strengthen intrusion prevention measures.

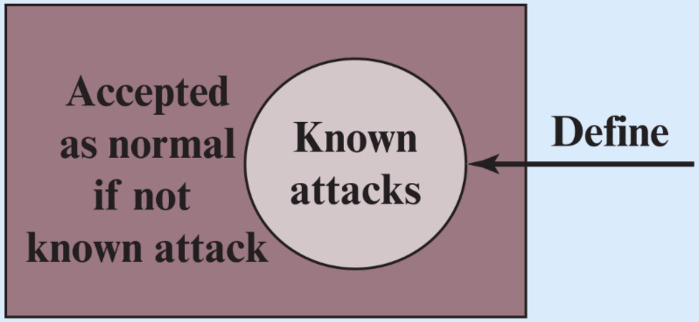
Effectiveness of an IDS

- Practically an intrusion detection system needs to detect a substantial percentage of intrusions while keeping the false alarms rate at acceptable level.
  - if too few intrusions detected → false security
  - if too many false alarms → ignore / waste time while analyzing the false alarm

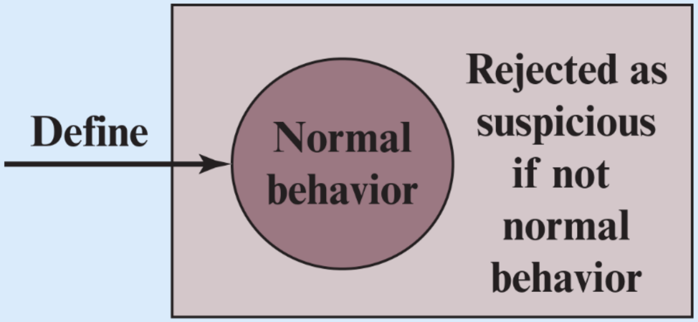


IDS Analysis Approaches

Two main approaches to analyze sensor data:



Signature based detection



Anomaly based detection

Signature Based Detection

- Examine the sensor data, looking for already known intruder attack patterns (signatures).
- The question of what information is relevant to an IDS depends upon what it is trying to detect.
  - e.g. attacks against DNS, FTP, IMAP etc.

- New attack patterns must continually be added to the IDS's database of signatures
- Novel attacks (with not previously known signatures) are NOT detected.

Note: Do not confuse it with cryptographic signatures

Signature examples

- MD5 hash of a malware file
- Specific byte sequences in email attachment indicating it is an executable
- Malicious commands executed by a process
  - e.g. `rm -rf /` in unix to delete all files
- Protocol abnormalities in packets
  - e.g. TCP packets with both SYN and FIN flags set
- Web packets going to URL of a known phishing sites, or accessing URL commonly associated with malwares
  - e.g. substring ‘phf’ in http request ‘GET /cgi-bin/phf?’, identifies those network packets as vehicles of an attack.

Note: Do not confuse it with cryptographic signatures

Anomaly Based Detection

- Involves a collection of information about legitimate user’s behavior over a period of time. Then, statistical tests or machine learning techniques are applied to observe them.
- Anything distinct from the usual behavior is assumed to be an intrusion activity.
  - e.g. flooding a host with lots of packet.

- Primary strength of anomaly-based approach is its ability to recognize novel attacks.
- But such IDSes can generate many false alarms and hence reduce their effectiveness.

Example metrics to observe

- Number of logins during an hour
- Number of times a command executed
- Number of outgoing messages
- Length of time between two events e.g. successive logins
- Quantity of resources (CPU, memory, printer etc.) used

Rule-based heuristic identification

A third approach is also sometimes used

- create rules for the normal usage pattern
- rules can be made based on
  - – expected behavior (ask system administrator)
  - – experience (by analyzing historical audit records)
  - – well known attacker behaviours (reported in research)
- any deviation from the baseline is considered intrusion
- like statistical anomaly detection, it does not require prior knowledge of security flaws

Rule-based heuristic identification

- it requires a large database of rules to be effective
  - rules are static, and need to be updated manually
- unlike anomaly based detection, a training phase is not required
- Examples rules:
  - Users should not be logged in more than one session
  - Users do not make copies of system, password files
  - Users who log in after hours should access the same files they used earlier

Where to install an IDS?

Host-based IDS:

- Monitors the characteristics of a single host and the events occurring within that host for suspicious activity.

Network-based IDS (NIDS):

- Monitors network traffic for particular network segments or devices and analyzes network, transport, and application protocols to identify suspicious activity.

Host/Applications based IDS

- Watch the host OS or the application logs in the audit information.

- These audit information includes events like the use of identification and authentication mechanisms (logins etc.), file opens and program executions, admin activities etc.

- This audit is then analyzed to detect trails of intrusion.

Host based IDS: Drawbacks

- The kind of information needed to be logged in is a matter of experience.

- Unselective logging of messages may greatly increase the audit and analysis burdens.
- Selective logging runs the risk that attack manifestations could be missed.

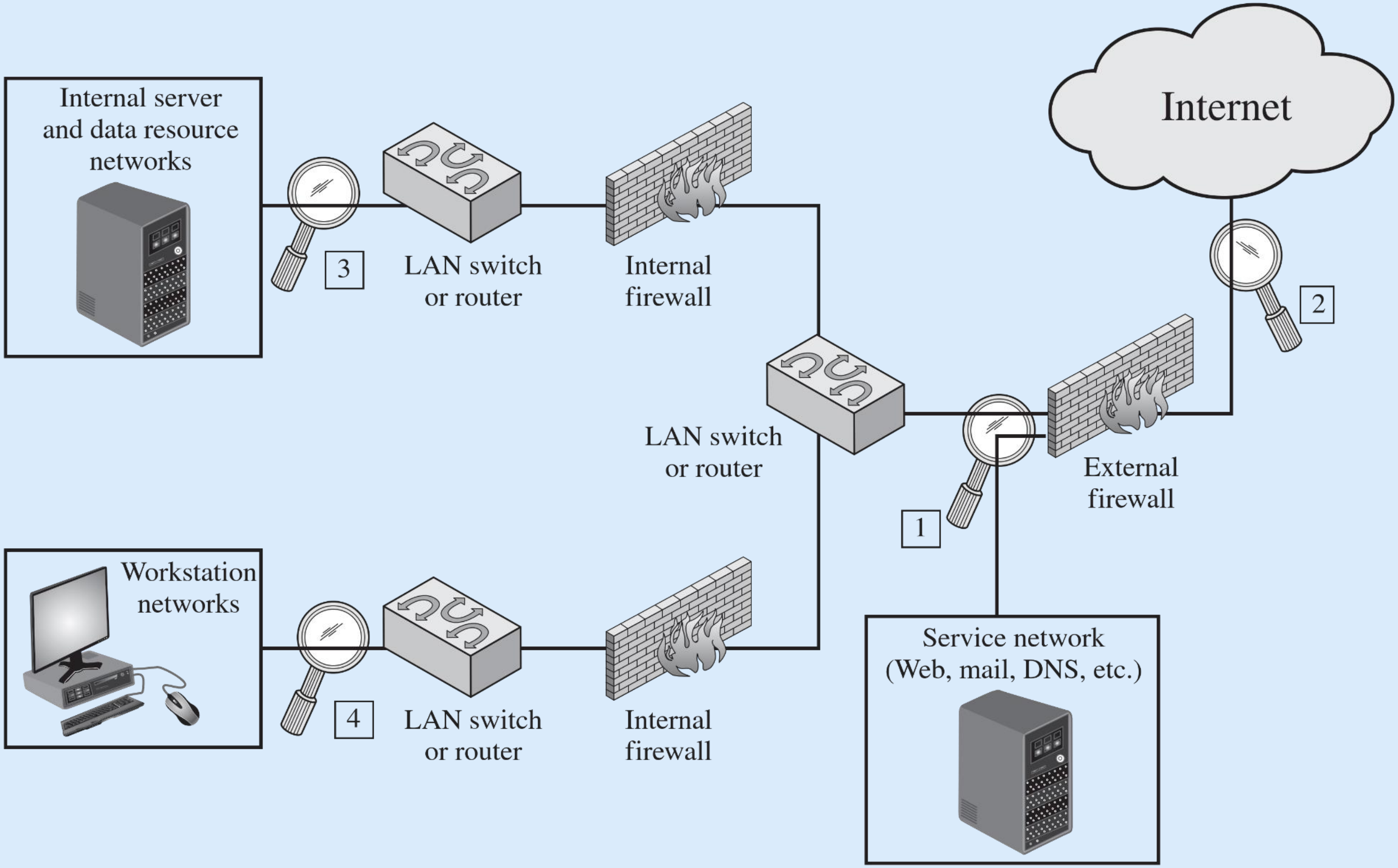
Host based IDS: Strengths

- Can determine exactly which processes and user accounts are involved in a particular attack
- Can see the exact intended outcome of an attack attempt
- Can detect both external and internal intrusions
- Helps in monitoring key components
- Near real-time detection and response
- No additional hardware needed

Network based IDS

- A network-based IDS monitors traffic at selected points on a network or interconnected set of networks.
- It examines the traffic packet by packet in real time or close to real time in order to detect intrusion patterns.
- A filter is usually applied to determine which traffic will be discarded or passed on to an attack recognition module. This helps to filter out known non-malicious traffic.

NIDS Sensor Placement



Network based IDS: Strengths

- Cost of ownership reduced
  - no per-host license
- Packet analysis
- Real time detection and response
- Malicious intent detection
- Operating system independence

Honeypots

- Decoy systems that designed to lure a potential attacker away from critical systems
  - An asset that solely exists to be attacked
- It could be an individual item, a system or entire network
- It could be a real system or emulated.

Purpose

- Divert an attacker from accessing critical systems
- Collect information about the attacker’s activity
  - Helps in IDS learning about new threats
- Engage the attacker to stay on the system long enough for administration to respond

Honeypots

Decoy Files

- Used as a "marker"
- In case of an access, read, copy, or deletion, it serves as an alert to monitors
- It could be anything: file, database, picture, email, account, etc.
- Normally used to deliver bogus information to attackers

Honey net

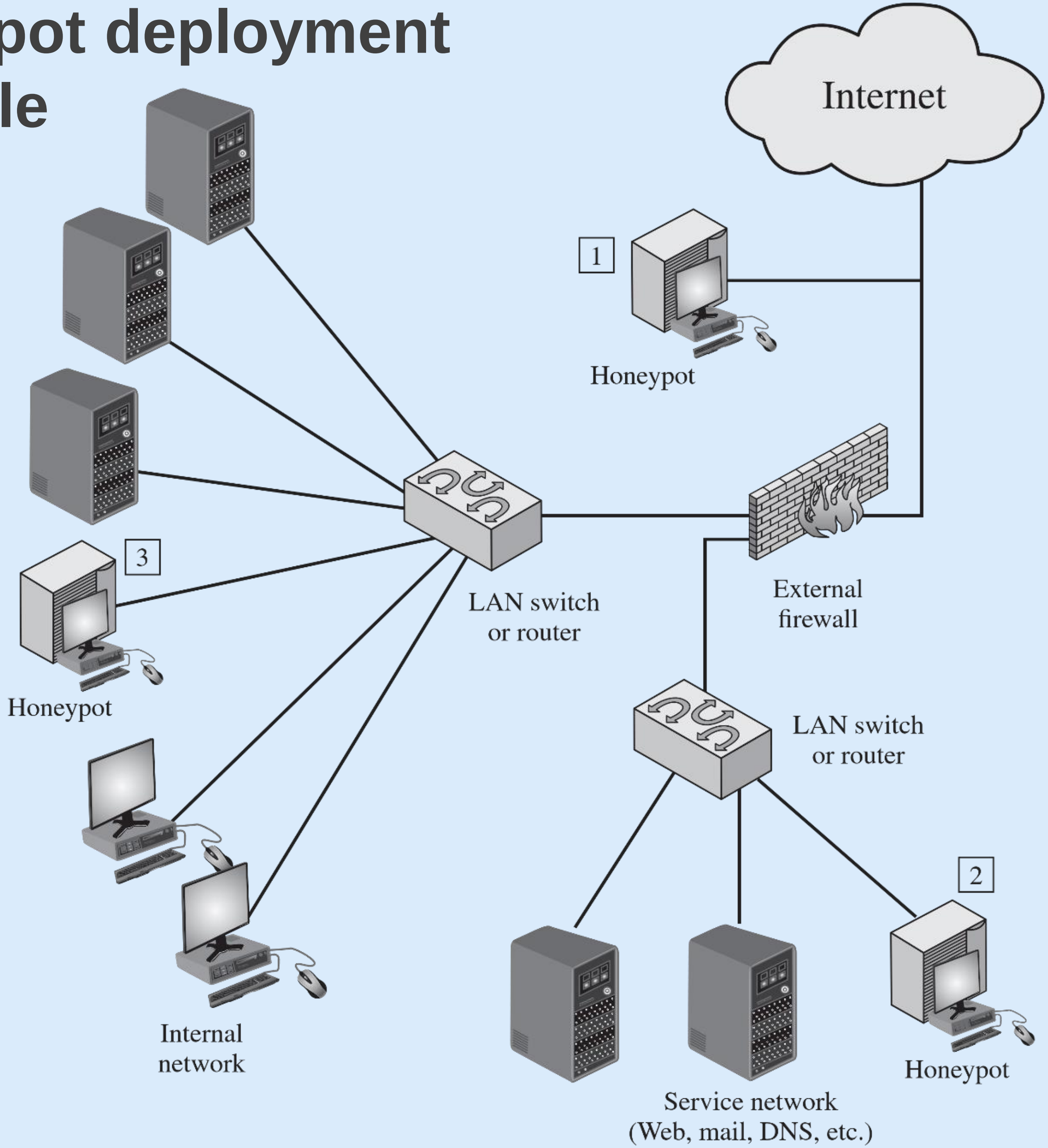
- Collection of two or more honeypots/decoy devices
- Could be at the same location or distributed
- Managed by same entity

Honeypot Interaction Level

It is the capability to mimic a real asset or object.

- High – More realistic that mimics real, legitimate computer or device with applications, activity, and changing content
  - Needed for more hacker interaction, intent, etc.
  - More involved setup and maintenance
- Low – Does very little to mimic real, legitimate device
  - Usually just TCP/IP port advertising or basic logon prompts
  - For early warning honeypots. Quicker setup, less ongoing maintenance, less risk
- If you can actually logon to a decoy, then you’re at least at Medium interaction

Honeypot deployment example



Deception Technology

- Honeypots are limited in scope
  - It uses static decoys which adversary starts to understand
  - requires expensive resources to implement and maintain
- Deception technology is a proactive cyber defense system through the use of decoys to lure, detect and defend, without the issues of scalability, skills and available resources.
  - Uses automated dynamic traps generated by AI
  - Immediate alerts with minimum false positive rates
  - Deploy traps according to the behavioral patterns of the hacker
  - Provide detailed reports for post cyber defense investigation