BAHRIA UNIVERSITY
Discovering Knowledge

Intrusion Detection Systems

Information Security (CSC-407)

Fall 2024 (BSE-7A & 7B)

Intruders

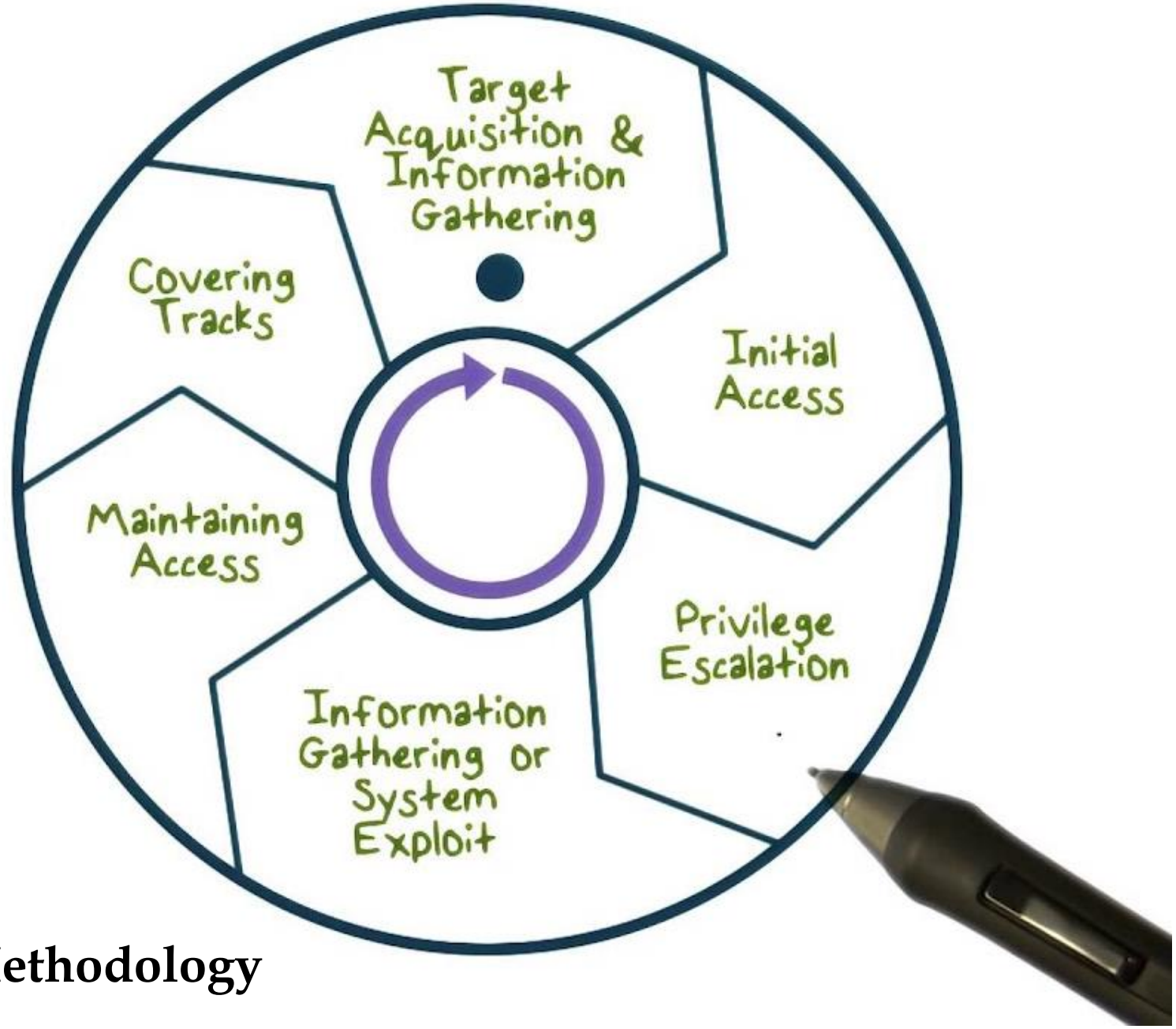
Intruders

- A significant security problem for **networked systems** is unwanted trespass by **users** or **software**.
- **User trespass** can take the form of:
  - *Unauthorized login*
  - *Acquisition of privileges*
  - *Performance of actions beyond authorization*

- **Software trespass** can take form of virus, worm or other malwares.

Intruder Behavior

- **Techniques** and **behavior patterns** of intruders are constantly shifting to exploit newly discovered weaknesses and to evade detection and countermeasures.
- However, intruders typically use steps from a **common attack methodology**.
- Helps security professionals to understand **adversary's tactics** beforehand.



Common Attack Methodology

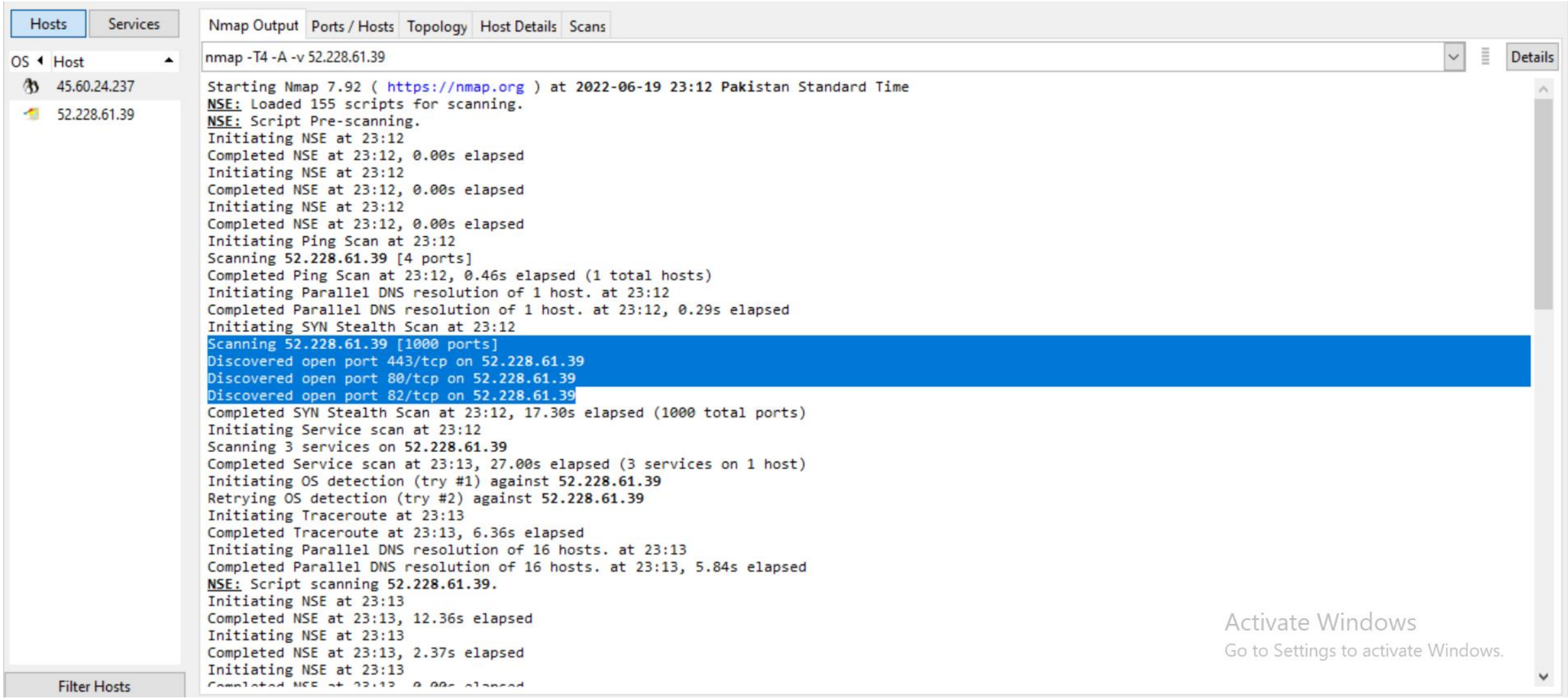
Intruder Behavior (Cont.)

1. **Target Acquisition and Information Gathering *(Footprinting and Reconnaissance)*:**
   - Attacker **identifies** and **characterizes** the target systems using publicly available information, both technical and non-technical.
   - Attacker uses **network exploration tools** to map target resources.
   - Reconnaissance Types:
     - ***Passive Reconnaissance:*** acquire information without directly interacting with target.
     - ***Active Reconnaissance:*** directly interacting with the target.

Intruder Behavior (Cont.)

- **Examples:**
  - ✓ Explore corporate website for information on corporate structure, personnel/key systems, details of web server and OS.
  - ✓ Gather information on target network using tools, such as ***DNS lookup, traceroute, etc***.
  - ✓ Map network for accessible services using tools such as ***NMAP (Network Mapper)***.
  - ✓ Send query email to customer service, review response for information on mail client/server, OS used and also details of person responding ***(Social Engineering / Phishing)***.

Intruder Behavior (Cont.)



Intruder Behavior (Cont.)

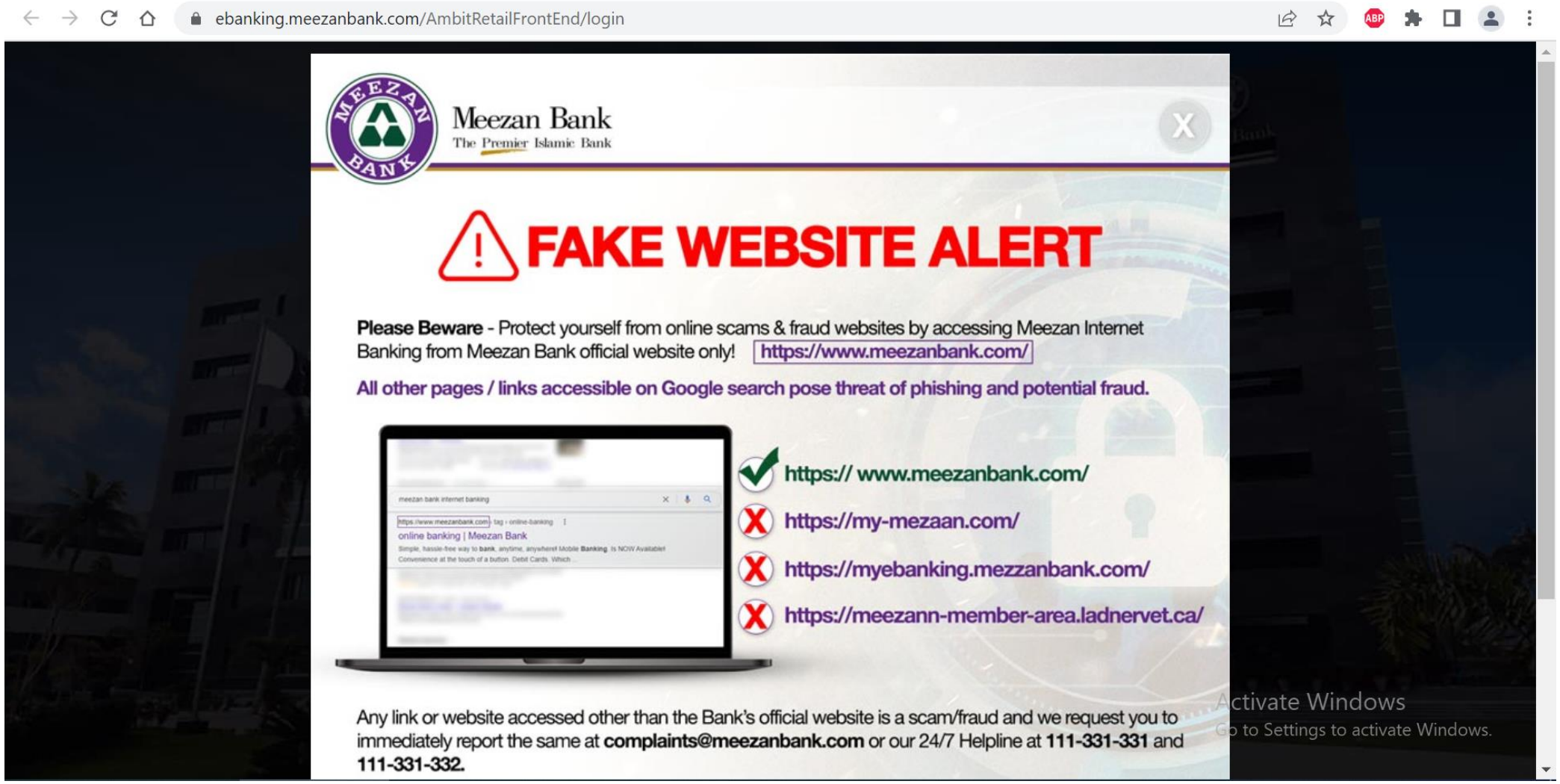
2. Initial Access:

- Attackers use vulnerabilities identified during reconnaissance / scanning phase to gain access to the target system / network.
- Access to a target system, typically by ***exploiting*** a remote network vulnerability, such as ***guessing weak authentication credentials or installation of malware***.

Intruder Behavior (Cont.)

- **Examples:**
  - ✓ Brute force/guess a password for user's web content management system.
  - ✓ Exploit vulnerability in web CMS plugin.
  - ✓ Send phishing emails with link to web browser exploit.
  - ✓ Gather information through fake websites.

Intruder Behavior (Cont.)



Intruder Behavior (Cont.)

3. Privilege Escalation:

- Actions taken on the system to *increase the privileges* available to the attackers to enable their desired goals on the target system.
- **Examples:**

✓ Exploit any vulnerable application to gain elevated privileges.

✓ Install sniffers to capture administrator passwords and use it to access privileged information.

Intruder Behavior (Cont.)

4. Information Gathering or System Exploit:

- *Access* or *modify* information or resources on the system.
- Navigate to another target system.
- **Examples:**

✓ Scan files for desired information.

✓ Transfer large numbers of documents to external repository.

✓ Use guessed or captured passwords to access other servers on network.

Intruder Behavior (Cont.)

5. Maintaining Access:

- Enable ***continued access*** by the attacker after the initial attack, such as by:
  - ➢ Installation of **backdoors** or other malicious software.
  - ➢ Addition of **covert** authentication credentials.
  - ➢ Other configuration changes to the system.

Intruder Behavior (Cont.)

- **Examples:**
  - ✓ Install Remote Administration Tool (RAT) with backdoor for later access.
  - ✓ Install rootkits at kernel level to gain full administrative access to the target computer.
  - ✓ Use administrator password to later access network.
  - ✓ Modify/disable anti-virus, firewall or IDS programs running on system.

Intruder Behavior (Cont.)

6. Covering Tracks:

- Attacker ***disables*** or edits ***audit logs*** to remove evidence of attack activity.
- Attacker ***uses rootkits*** and other measures to hide covertly installed files or code.
- **Examples:**

✓ Use rootkit to hide files installed on system.

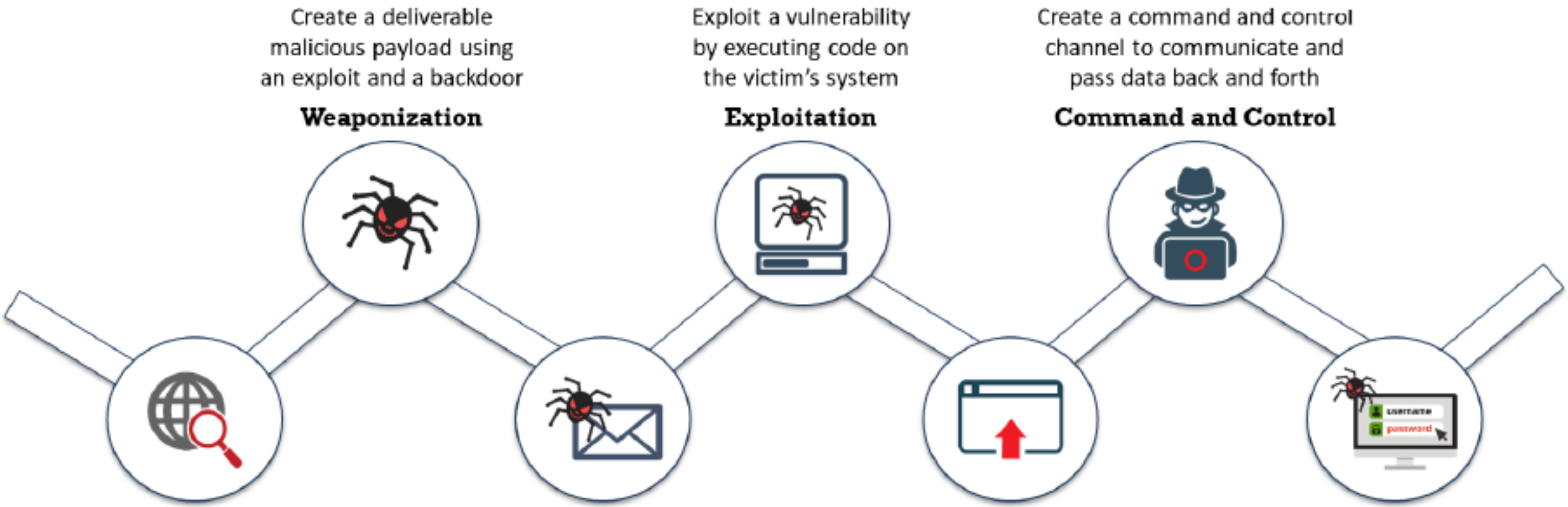
✓ Edit log files to remove entries generated during the intrusion.

Cyber Kill Chain Methodology

- **Cyber Kill Chain Methodology;** a method by **Lockheed Martin** based on concept of ***military kill chains****.
- **Cyber Kill Chain Methodology is** another framework that helps security professionals to understand the adversary's tactics, techniques, and procedures beforehand.
- A **seven-phase** protection mechanism to mitigate and reduce cyber threats.

****A concept that originated within the United States military to model the sequential steps that must be conducted in order to successfully plan and deliver an attack and deliver a "kill" (delivery of the objective and destruction of a target).***

Cyber Kill Chain Methodology (Cont.)

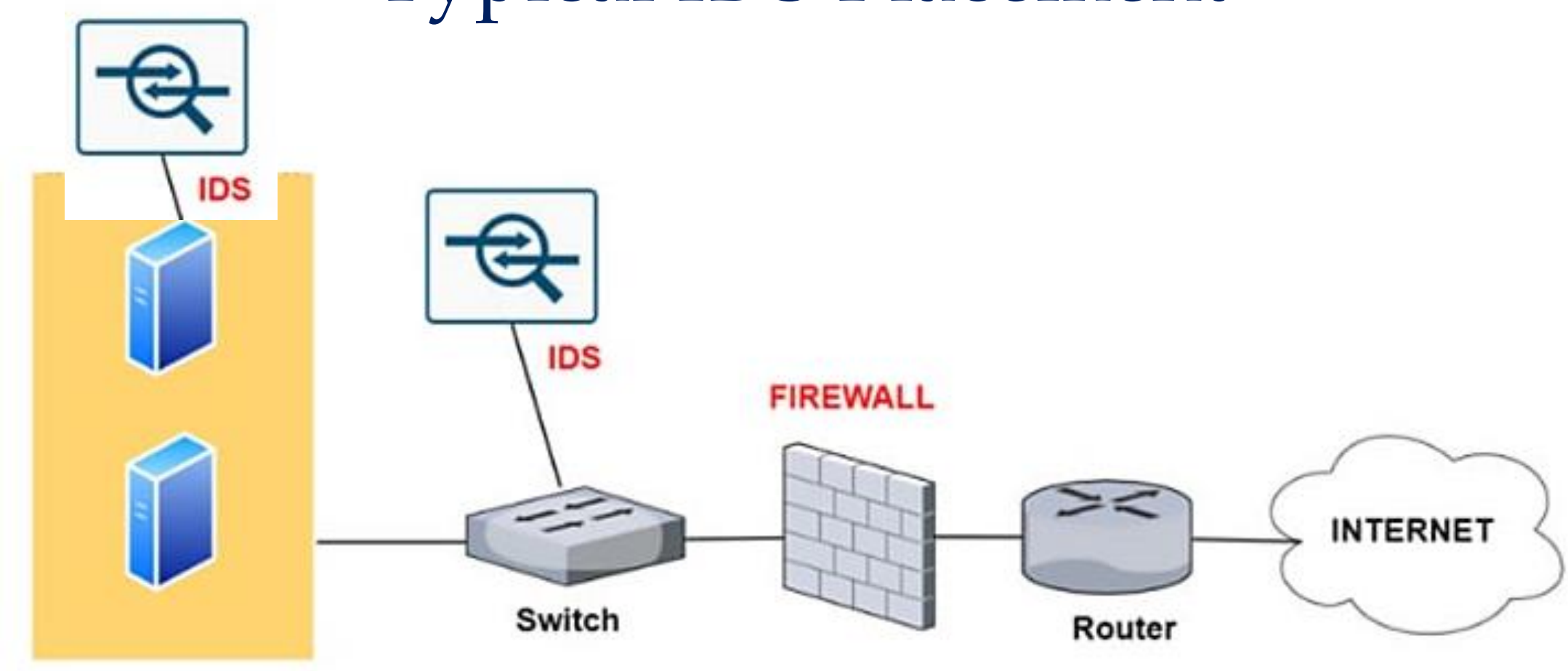


Intrusion Detection System

Intrusion Detection

- **Intrusion Detection:** a **security service** that ***monitors*** and ***analyzes*** system events for providing ***real-time warning*** of attempts to access system resources in an unauthorized manner.
- **Intrusion detection** is a line of defense which has been the focus of much research in recent years.
- **Note: authentication**, **access control** and **firewalls** all play roles in countering intrusions.

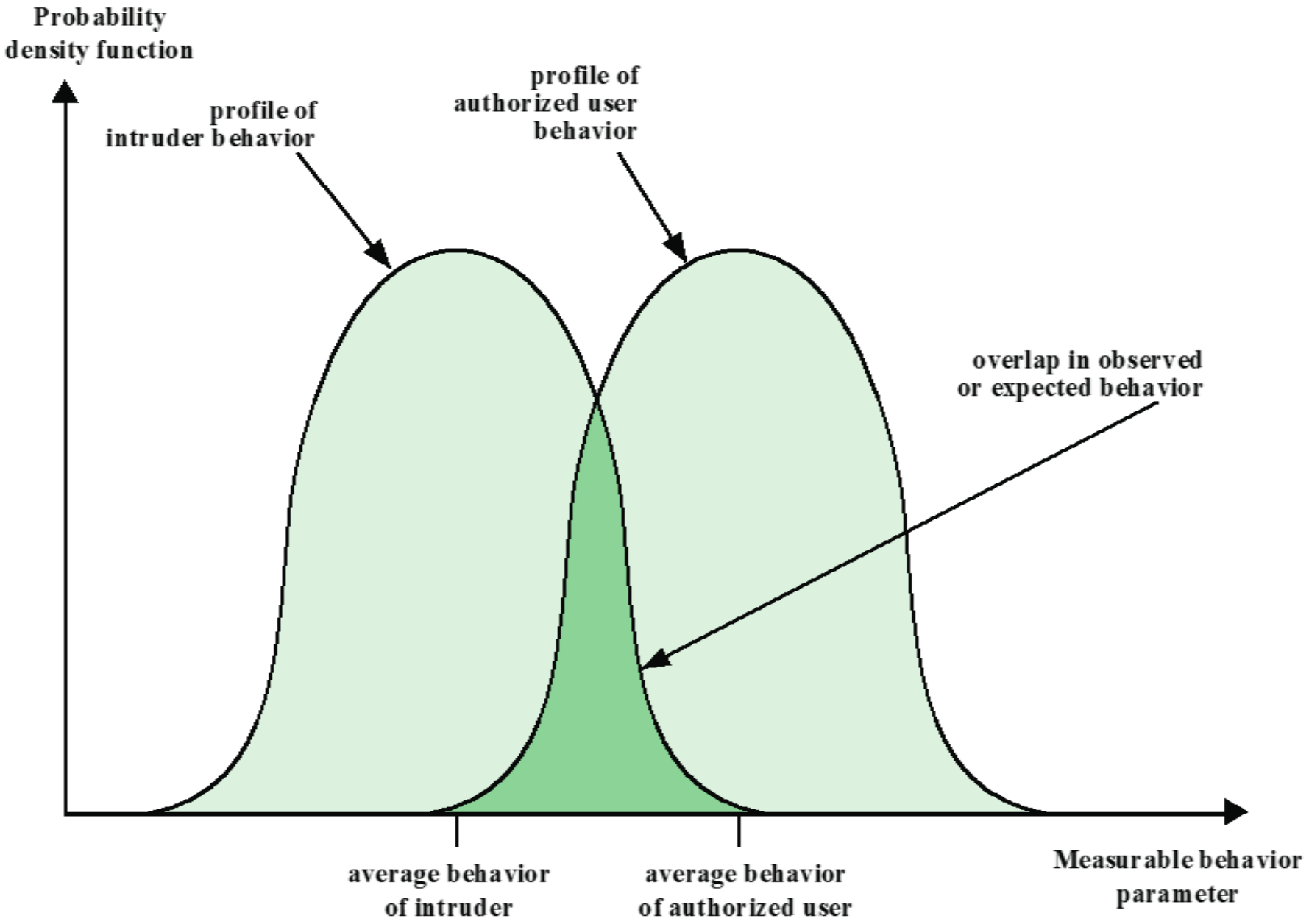
Typical IDS Placement



Intruder vs. Legitimate User Behavior

- Intrusion detection is based on the **assumption** that behavior of an **intruder** differs from that of a **legitimate user** in ways that can be **quantified**.
- However, we cannot expect that there will be an exact distinction between an **intruder** and an **authorized user**.
- Though the typical **behavior** of an intruder differs from the typical behavior of an authorized user, there is an **overlap** in these behaviors as well.

Profiles of Behavior of Intruders and Authorized Users



False Detection

- In general, the IDS cannot provide accurate detection. Hence, the alarms generated by an IDS can be categorized as:
  1. **False positives (false alarms)**: authorized users are identified as intruders ***OR*** benign activity is identified as malicious.
  2. **False negatives:** intruders are not identified as intruders ***OR*** malicious activity is failed to be identified.
- ***Note:*** while every **intrusion** is an **incident**, not every **incident** is an **intrusion**.

False Detection

- It is not possible to eliminate both **false positive** and **false negative**, as reducing one of the errors has an effect of increasing the other.
- It is intuitive that organizations prefer to reduce the **false negatives** at the cost of an increased **false positive**.

IDS / IPS

- **Intrusion detection systems (IDS)** and **intrusion prevention systems (IPS)** are designed to **detect** and **counter** different threats.
- **IDS / IPS** can be reasonably effective against ***known and less sophisticated attacks***. However, **IDS / IPS** are likely to be less effective against ***more sophisticated and targeted attacks***. **Why?**
- The sophisticated attacks are more likely to use ***new***, ***zero-day exploits*** and ***attackers can better hide their activities*** on the targeted system.

IDS / IPS (Cont.)

- Hence **IDS / IPS** need to be part of a **defense-in-depth** strategy. This may include:
  - Detailed audit trails
  - Application level security
  - Encryption of sensitive information
  - Active management of operating system
  - Strong authentication and authorization controls

IDS Components

An IDS comprises three logical components:

1. Sensors:

- Responsible for collecting data.
- Input for a sensor may be any part of a system that could contain evidence of an intrusion.
- Types of input to a sensor includes ***network packets***, ***log files***, and ***system call traces***.
- Sensors collect and forward this information to the **analyzer**.

IDS Components (Cont.)

An IDS comprises three logical components (Cont.):

2. **Analyzers:**
   - Receive input from one or more sensors or from other analyzers.
   - Responsible for **determining** if an intrusion has occurred.
   - Output is an **indication** that an intrusion has occurred.
3. **User interface:**
   - The user interface enables a user to view output from the system or control the behavior of the system.

IDS Analyzer

Analysis Approach

- **Signature detection:** uses a set of known *malicious data patterns (signatures)* that are compared with current behavior to decide if is that of an intruder.
- **Anomaly detection:** involves the collection of data relating to *behavior of legitimate users* over a *period of time*.

Signature Detection

- Signature detection approaches **match** a large collection of ***known patterns*** against data stored on a system or in transit over a network.
- Signature approaches detect intrusion by observing events in the system and applying a **set of signature patterns** to data.
- **E.g.** ***bytes flow / size*** in network traffic ***OR*** a ***series of instructions***.

Signature Detection (Cont.)

- Signature detection approaches **directly** define ***malicious / unauthorized*** behavior, hence can quickly and efficiently identify known attacks.
- Signatures need to be large enough to minimize the **false alarm rate**, while still detecting a sufficiently large fraction of malicious data.
- However, signature approaches can only identify **known attacks** for which it has patterns.

Signature Detection (Cont.)

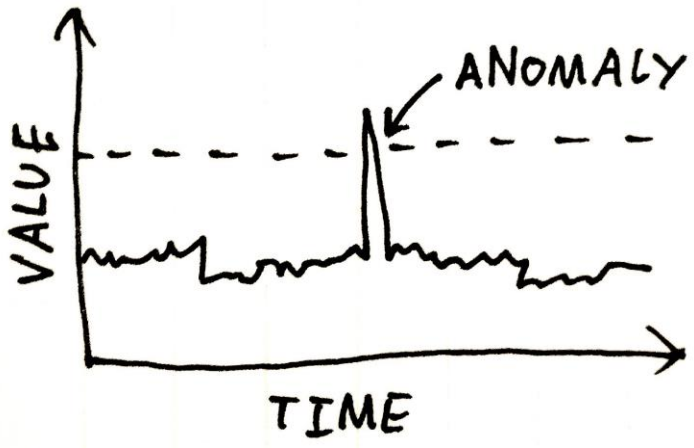
- **Advantages:**
  - Relatively low cost in-terms of time and resource usage
  - Wide acceptance
- **Disadvantages:**
  - Significant effort required to constantly identify and review new malware to create signatures able to identify it
  - Inability to detect zero-day attacks

Anomaly Detection

- Anomaly approaches aim to define ***normal or expected behavior*** to identify ***malicious or unauthorized behavior (i.e. variation from the norms)***.
- Involves first **developing a base-line model** of legitimate user behavior by collecting and processing sensor data from the normal operation of the monitored system in a **training phase**.
  - ***This may occur at different times, or there may be a continuous process of monitoring and evolving the model over time.***

Anomaly Detection (Cont.)

- Once this model exists, the current observed behavior is analyzed to **classify** if the behavior is of a legitimate user or of an intruder.
- Only anomaly detection is able to detect **unknown, zero-day attacks**, as it starts with known good behavior and identifies anomalies to it.



Anomaly Detection (Cont.)

- A variety of classification approaches are used which can be broadly categorized as:

Statistical	Knowledge based	Machine-learning
• Analysis of the observed behavior using **univariate**, **multivariate** or **time-series** models of observed metrics.	• Approaches use an **expert system** that classifies observed behavior according to a set of rules that model legitimate behavior.	• Approaches determine a suitable classification model from the training data using data mining techniques.

Anomaly Detection (Cont.)

Statistical Approaches:

- Use the captured sensor data to develop a **statistical profile** of the observed metrics.
- In **univariate** models, each metric is treated as an **independent** variable.
- **Multivariate** models consider **correlations** between metrics.
- **Time-series** models use the **order** and **time difference** between observed events to better classify the behavior.

Anomaly Detection (Cont.)

Statistical Approaches (Cont.):

➢ Advantages:

- Relative simplicity
- Low computation cost

➢ Disadvantages:

- Difficulty in selecting suitable **metrics** to obtain reasonable balance between **false positives** and **false negatives**
- **Not all behaviors** can be modeled using statistical approaches

Anomaly Detection (Cont.)

Knowledge-Based Approaches:

- Classify the observed data using a ***set of rules***.
- These rules are developed, ***usually manually***, during the training phase to characterize the observed training data into ***distinct classes***.
- Rules and classes are then used to classify the observed data in the detection phase.

Anomaly Detection (Cont.)

Knowledge-Based Approaches (Cont.):

- **Advantages:**
  - Flexibility *(upto the rule maker experience)*
- **Disadvantages:**
  - Need for *human experts* to assist with development of rules
  - Difficulty and much time required to develop a *high-quality knowledge / rules* from data

Anomaly Detection (Cont.)

Machine-Learning Approaches:

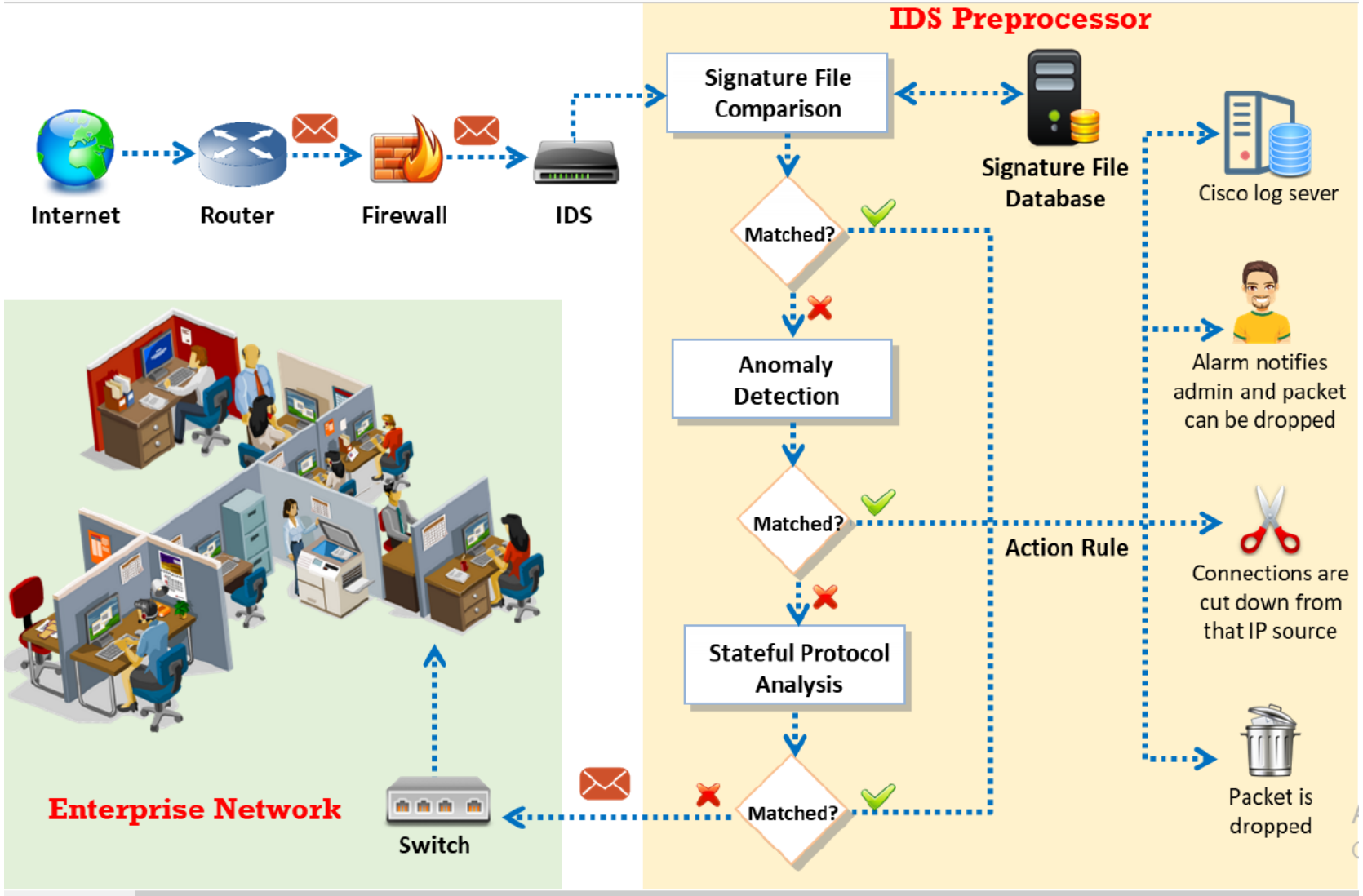
- Use **data mining techniques** to develop a model using the labeled normal training data.
- This model is then able to classify subsequently observed data as either normal or anomalous.
- A variety of machine-learning approaches have been tried, with varying success, such as ***Bayesian networks, Neural networks, Genetic algorithms, etc.***

Anomaly Detection (Cont.)

Machine-Learning Approaches (Cont.):

- **Advantages:**
  - Adaptability
- **Disadvantages:**
  - Model generation typically requires *significant time* and *high computational resource* cost
  - Dependency on *training data*

Signature & Anomaly

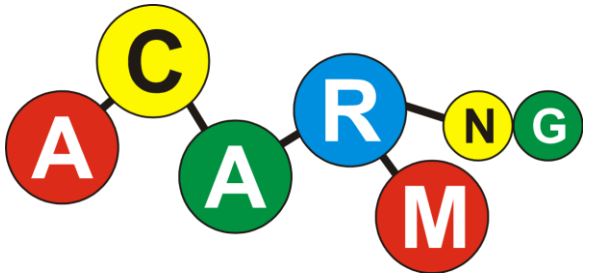


Signature & Anomaly (Cont.)

- In **signature-based IDS**; network traffic is checked with any of the signatures in the ***signature file database***.
  - ➢ If no match found → forward to Anomaly-based IDS.
- In **anomaly-based IDS**; statistical/knowledge based/ML techniques is used to compare monitored traffic with the ***normal traffic profile***.
  - ➢ If no match found → forward to Stateful Protocol Analysis.

Signature & Anomaly (Cont.)

- If **match found** in any of three stages of intrusion detection, then possible actions include:
  - *Disconnect connection from source IP*
  - *Drop packet*
  - *Log activity*
  - *Generate an alarm*
- If **no match found** in any of three stages of intrusion detection, then *pass to the destined network*.



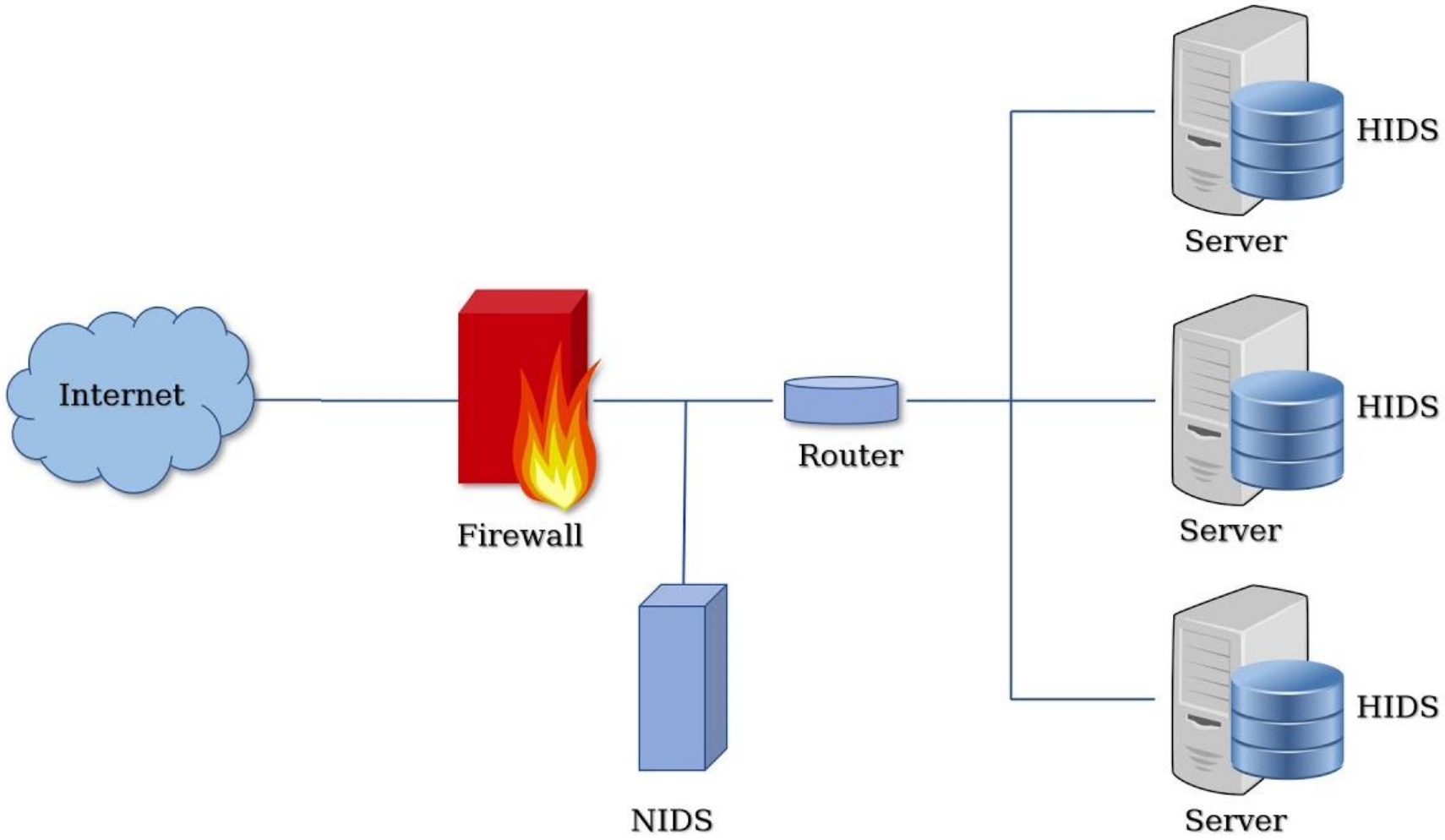
HIDS/NIDS

VERISYS

IDS Classification

- **IDSs** can be classified based on the **source of data** analyzed:
  - **Host-based IDS (HIDS):** monitors the characteristics of a **single host** and the events occurring within that host, such as **process identifiers** and **system calls**.
  - **Network-based IDS (NIDS):** monitors **network traffic** for particular network segments or devices and analyzes different **protocols** to identify suspicious activity.

IDS Classification (Cont.)

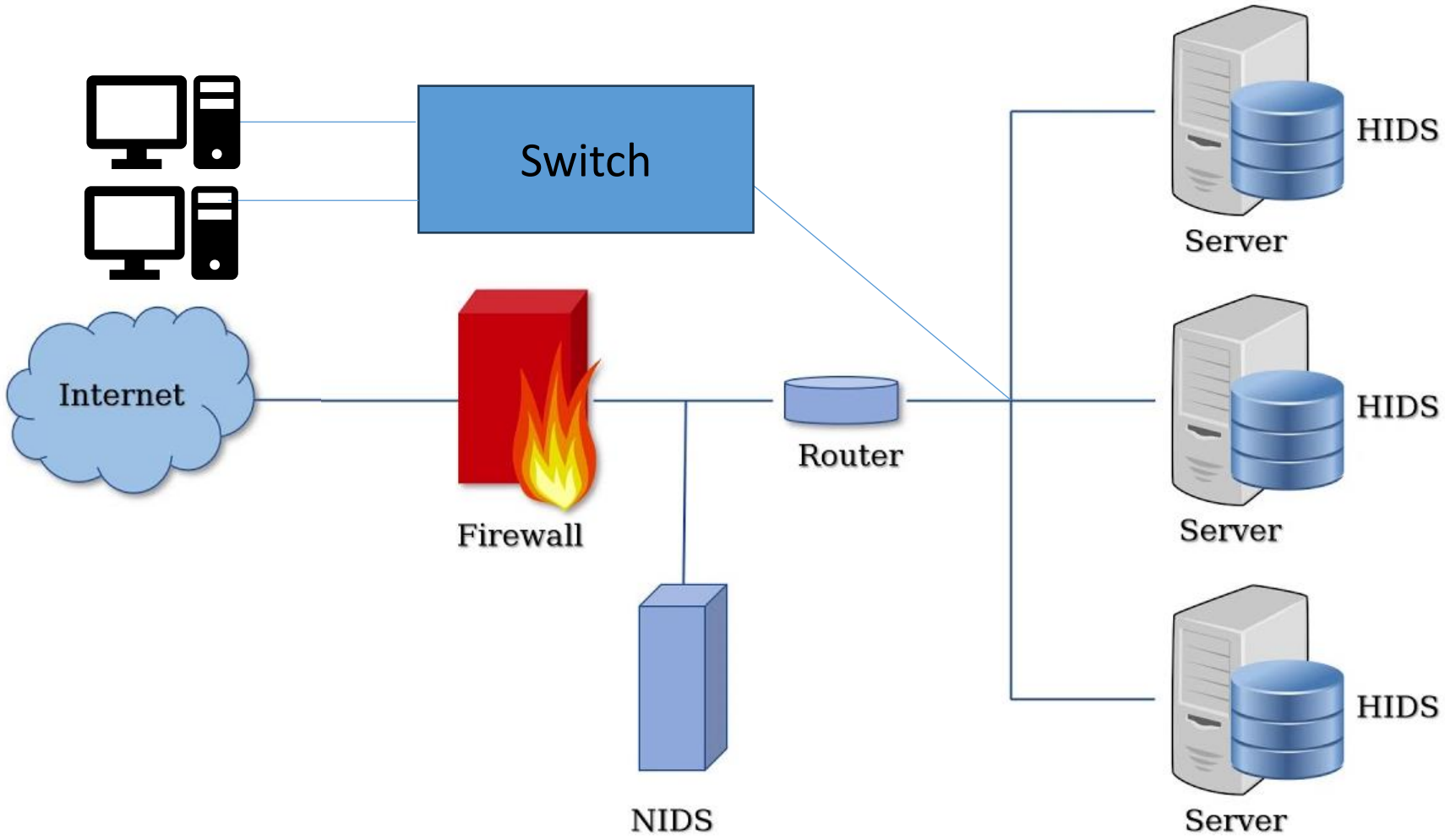


Host-based IDS

Host-Based IDS

- **HIDS** adds a layer of **security software** to vulnerable or sensitive systems, such as **database servers / file server**.
- The **HIDS** monitors activity on a particular system in a variety of ways to detect suspicious behavior.
- **HIDS** examines **user** and **software activity** on a host.
- The primary benefit of a **HIDS** is that it can detect **internal intrusions**, something that is not possible with **NIDS**.

Host-Based IDS (Cont.)

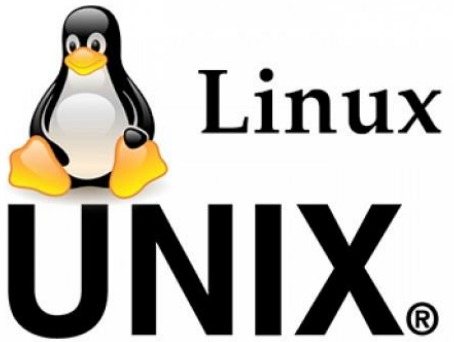


Host-Based IDS (Cont.)

- **Common data sources in HIDS include:**
  - *System call traces*
  - *File integrity checksums (hash)*
  - *Audit (log file) records*
  - *Registry access*
- **HIDS** can use either **anomaly** or **signature** approaches to detect unauthorized behavior on the monitored host.

Anomaly HIDS

- The majority work on **anomaly HIDS** has been done on ***Unix*** and ***Linux*** systems due to the ease of gathering suitable data required for this purpose ***(e.g. open source)***.
- Windows have "traditionally" not used **anomaly HIDS**.



Anomaly HIDS (Cont.)

Anomaly HIDS through System Calls:

- Majority of the **anomaly HIDS** is based on **system call traces**.
- The **current process** behavior is examined using a suitable decision engine that analyzes **system call traces**.
- **E.g.** performing **write()** operations on read-only file where only **read()** operation should be performed.

Anomaly HIDS (Cont.)

Anomaly HIDS through System Calls (Cont.):

- Initial works compared the observed sequences of **system calls** with sequences from the **training phase** to obtain a **mismatch ratio** that determines whether the sequence is normal or not.
- Later work used **Machine Learning** algorithms to make classification. These algorithms are reported to provide reasonable intruder detection rates of **95-99%** while having **false positive rates** of less than **5%**.

Anomaly HIDS (Cont.)

Anomaly HIDS through System Calls (Cont.):

- *Examples of Ubuntu Linux System Calls*: ***accept, access, acct, adjtime, aiocancel, aioread, aiowait, aiowrite, alarm, async_daemon, auditsys, bind, chdir, chmod, chown, chroot, close, connect, creat, dup, dup2, execv, execve, exit, ETC...***
- **Advantage: System call traces** provides the richest information source for a **HIDS**.
- **Disadvantage:** it imposes a moderate load on the monitored system to gather and classify data.

Anomaly HIDS (Cont.)

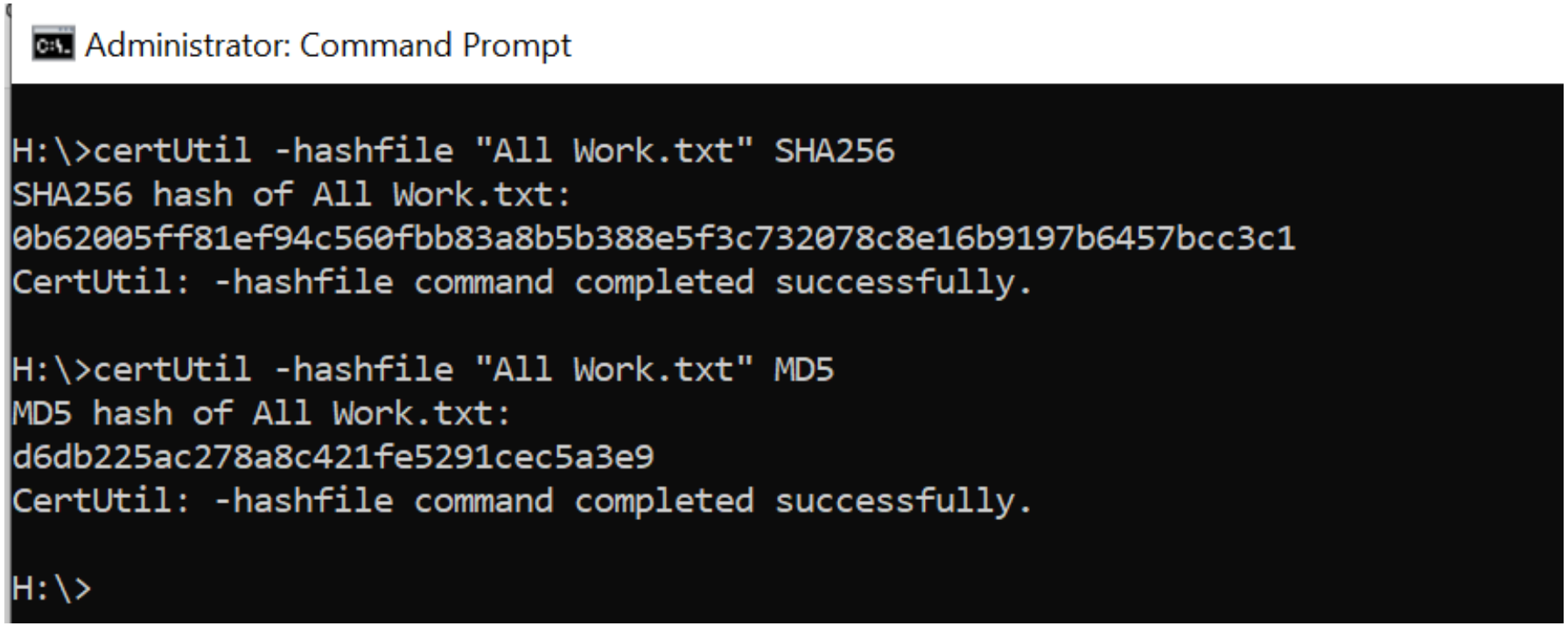
Anomaly HIDS through Monitoring of Files:

- An alternative to examining ***current process*** behavior is to look for changes to ***important files*** on monitored host.
- **HIDS** benchmark and monitors the status of **key system files** and detect when an intruder creates, modifies, or deletes monitored files.
- **HIDS** uses a ***cryptographic checksum*** to check for any changes from the known good baseline for the monitored files.
- Typically, all **program binaries, scripts** and **configuration** files are monitored, either on **each access** or on a **periodic scan**.

Anomaly HIDS (Cont.)

- Run as Administrator "cmd" and type:

certUtil -hashfile "All Work.txt" SHA256



Anomaly HIDS (Cont.)

Anomaly HIDS through Monitoring of Files (Cont.):

- **Advantage: very sensitive** to changes in the monitored files, as a result of intruder activity or for any other reason.
- **Disadvantage:**
  - Difficulty in determining **which files to monitor**.
  - Difficulty in having access to a **known good copy** of each monitored file **to establish baseline value**.
  - Difficulty in protecting the **database of file signatures**.

Signature HIDS

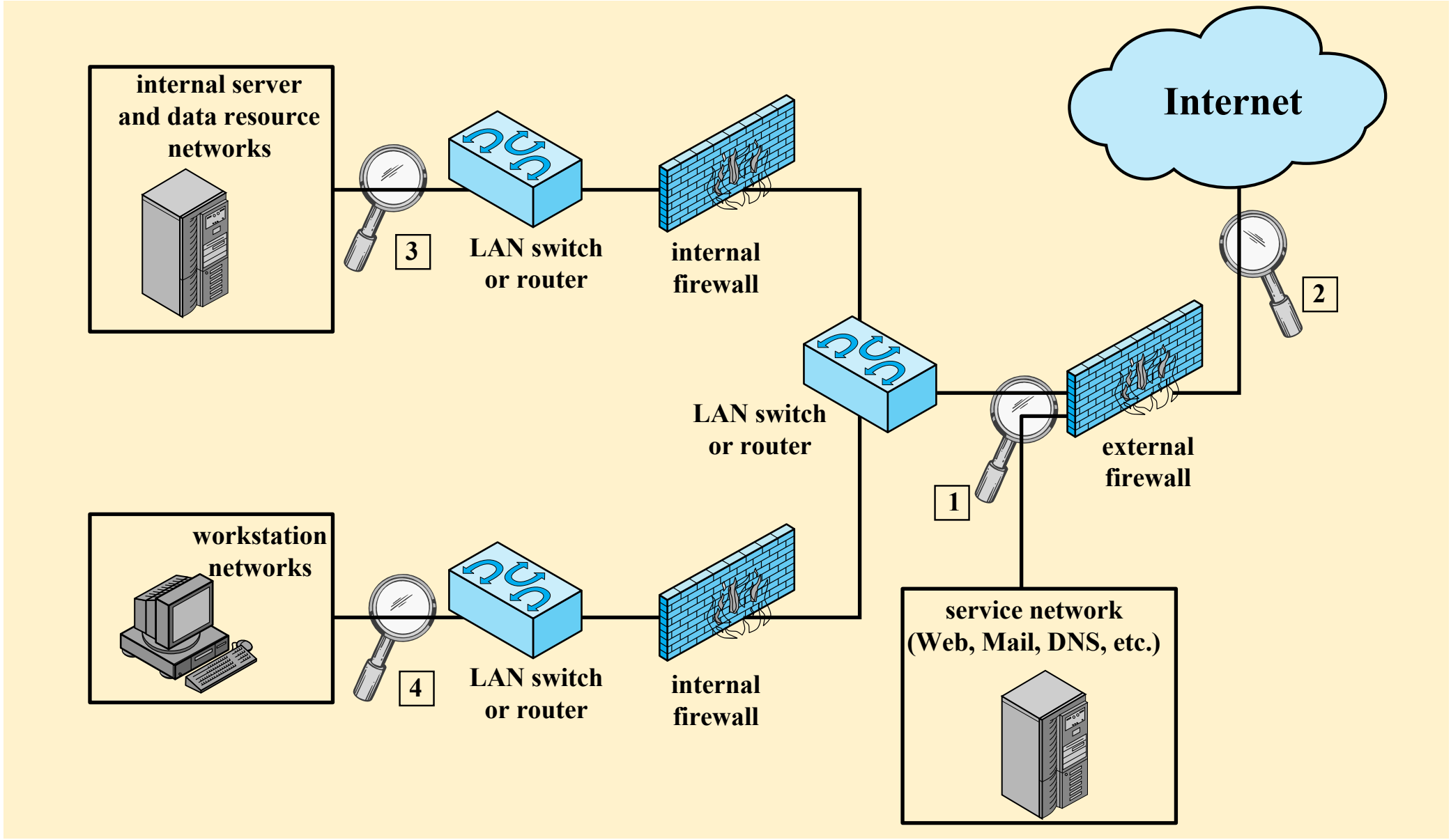
- The **signature HIDS** is widely used, particularly as seen in **anti-malware** products.
- These are commonly used on ***Windows systems***.
- They use a database of **file signatures** which are patterns of data found in known malicious software.
- Efficient at detecting **known malware**, but not capable of detecting **zero-day attacks** that do not have known signatures.

Network-based IDS

Network-Based IDS

- **NIDS** examines the traffic **packet by packet** in real time, or close to real time, to attempt to detect intrusion patterns.
- **NIDS** examines packet traffic directed toward potentially vulnerable systems on a network.
- **NIDS** may examine **network-**, **transport-**, and **application-**layer protocol activity.
- **NIDS** monitors traffic at **selected points** on a network. Typically in the ***border security infrastructure*** of an organization, either incorporated in or association with the **firewall**.

Example of NIDS Deployment



Network-Based IDS (Cont.)

- **NIDS** typically focus on monitoring for **external intrusion** attempts by analyzing both **traffic patterns** and **traffic content**.
- However, with the increasing use of **encryption**, **NIDS** have lost access to significant content, hence hindering their ability to function well.
- As with **HIDS**, **NIDS** also makes use of **signature detection** and **anomaly detection** approaches.

Anomaly NIDS

Type of attacks that anomaly NIDS can detect:

- **Denial-of-service (DoS) attacks:** to **overwhelm** the target system by either significantly increased packet traffic or significantly increase connection attempts.
- **Scanning:** an attacker probes a network or system by sending different kinds of packets. Using the responses received from the target, the attacker can learn many of the system's **characteristics** and **vulnerabilities**. Scanning can be detected by a typical **flow patterns** at different communication layers.

Anomaly NIDS (Cont.)

Type of attacks that anomaly NIDS can detect (Cont.):

- **Worms:** worms spread among hosts, where some worms propagate quickly and ***use large amounts of bandwidth***. Worms can also be detected because they can ***cause hosts to communicate with each other*** that typically do not. Also, worms can ***cause hosts to use ports that they normally do not use***.

Signature NIDS

Type of attacks that signature NIDS can detect:

- Signature-based NIDS are effective at detecting common and well-known attacks.
- However, it cannot detect new or unknown attacks.

Thank You!