

# Week 1 Task: Introduction to Network Security and Basic Security Practices

Intern: Hamaiz Siddiqui

## Task 1: Understanding Cybersecurity Fundamentals

Cybersecurity is essential in today's digital world, where information is constantly shared and stored online. To protect this information, we rely on several key concepts and practices. This summary will cover the CIA Triad, types of cyber attacks, and the importance of network security, data protection, and user authentication.

### CIA Triad: Confidentiality, Integrity, and Availability

The CIA Triad is a foundational model that guides cybersecurity policies and practices. It consists of three core principles:

**Confidentiality:** This principle ensures that sensitive information is accessible only to those authorized to view it. Techniques like encryption and access control are employed to maintain confidentiality.

**Integrity:** Integrity refers to the accuracy and trustworthiness of data. It ensures that information remains unaltered during storage and transmission. Hashing and digital signatures are common methods used to verify integrity.

**Availability:** This principle ensures that information and resources are accessible when needed. Organizations implement redundancy and disaster recovery plans to maintain availability, protecting against system failures and attacks.

 wallarm



## **Types of Cyber Attacks**

Cyber attacks can take many forms, each targeting different vulnerabilities. Here are some of the most common types:

**Phishing:** This is a technique used to trick individuals into providing sensitive information (like passwords or credit card numbers) by masquerading as a trustworthy source, often through emails or deceptive websites.

**Malware:** Short for "malicious software," malware includes viruses, worms, and ransomware that can damage, disrupt, or gain unauthorized access to a system. Malware often spreads through infected email attachments or downloads.

**Denial-of-Service (DoS):** In a DoS attack, attackers overload a system with traffic, causing it to become slow or unavailable. This disrupts services for legitimate users.

## **Importance of Network Security, Data Protection, and User Authentication**

### **Network Security**

Network security involves protecting the integrity and usability of networks and data. Measures include firewalls, intrusion detection systems, and secure access protocols. A robust network security strategy helps prevent unauthorized access and data breaches.

### **Data Protection**

Data protection focuses on safeguarding personal and sensitive information from loss, corruption, or unauthorized access. Techniques such as data encryption, backups, and compliance with regulations (like GDPR) ensure that data is secure and can be recovered if lost.

### **User Authentication**

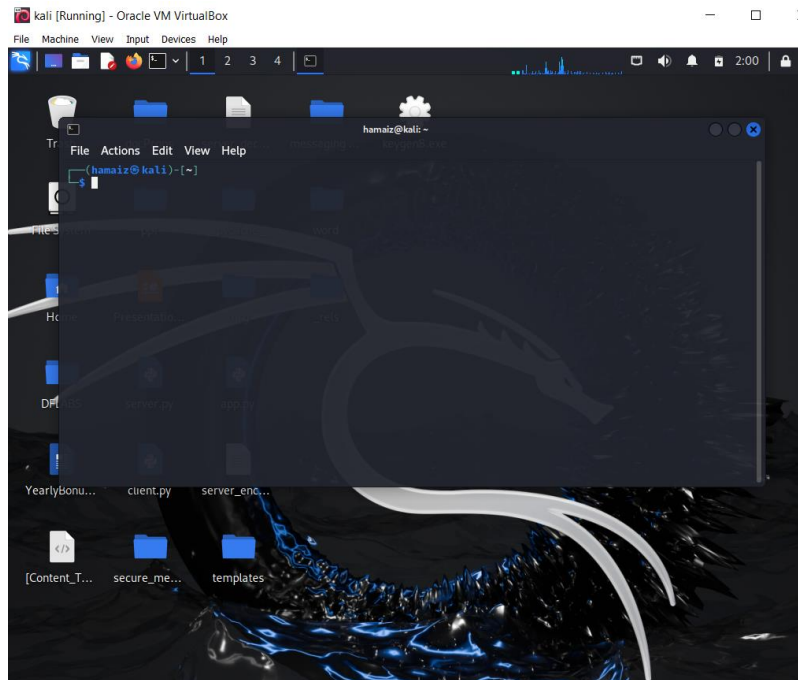
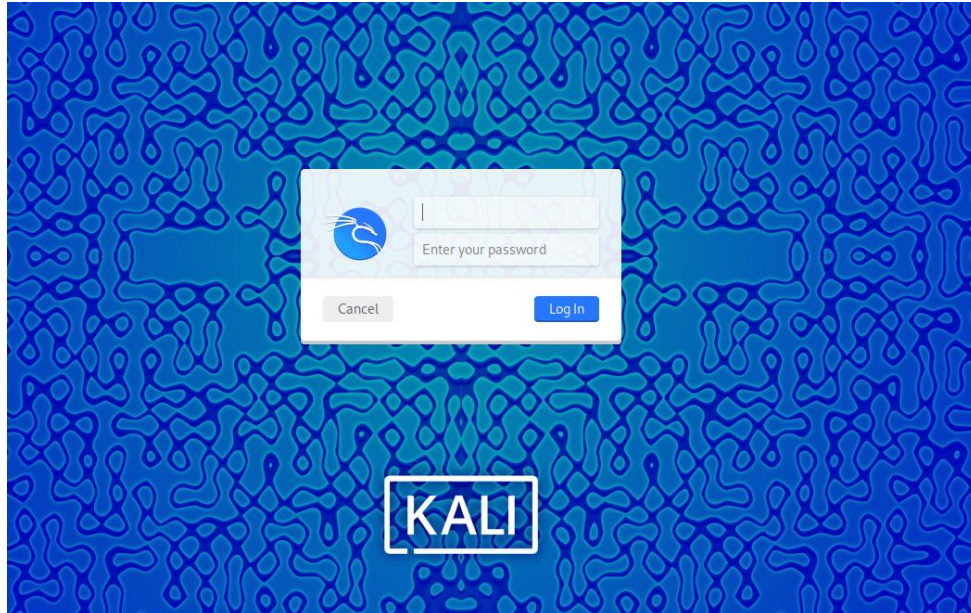
User authentication is the process of verifying the identity of users accessing a system. Strong authentication methods, such as multi-factor authentication (MFA), enhance security by requiring users to provide multiple forms of verification (like passwords and biometric data). This helps prevent unauthorized access, even if passwords are compromised.

## **Conclusion**

Understanding these fundamental cybersecurity concepts is crucial for protecting information and systems. By ensuring confidentiality, integrity, and availability, recognizing types of cyber attacks, and implementing strong security measures, individuals and organizations can safeguard their digital assets in an increasingly connected world.

## Task 2: Setting Up a Secure Virtual Environment

### VM SETUP



## Disable Root Login for SSH

```
File Actions Edit View Help
(hamaiz@kali)-[~]
$ sudo nano /etc/ssh/sshd_config
[sudo] password for hamaiz:
```

```
File Actions Edit View Help
GNU nano 8.0 /etc/ssh/sshd_config *

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin no prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2
```

```
(hamaiz@kali)-[~]
$ sudo systemctl restart ssh
```

## Configure Firewall Settings Using UFW

```
(hamaiz@kali)-[~]
$ sudo ufw enable
Firewall is active and enabled on system startup

(hamaiz@kali)-[~]
$ sudo ufw default deny incoming
Default incoming policy changed to 'deny'
(be sure to update your rules accordingly)

(hamaiz@kali)-[~]
$ sudo ufw default allow outgoing
Default outgoing policy changed to 'allow'
(be sure to update your rules accordingly)

(hamaiz@kali)-[~]
$ sudo ufw allow ssh
Rule added
Rule added (v6)

(hamaiz@kali)-[~]
$ sudo ufw allow http
Rule added
Rule added (v6)
Rule added
Rule added (v6)

(hamaiz@kali)-[~]
$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To Action From
--
22/tcp ALLOW IN Anywhere
80/tcp ALLOW IN Anywhere
443 ALLOW IN Anywhere
22/tcp (v6) ALLOW IN Anywhere (v6)
80/tcp (v6) ALLOW IN Anywhere (v6)
443 (v6) ALLOW IN Anywhere (v6)
```

# Task3: Network Security Basics: Packet Analysis

The image shows a Wireshark interface with a packet capture list and a packet details pane. The packet list shows various protocols including TLSv1.3, TCP, and User Datagram Protocol. The selected packet (No. 1) is a User Datagram Protocol packet from 10.0.2.15 to 172.217.169.234, port 53. The details pane shows the packet structure: Ethernet II, Internet Protocol Version 4, User Datagram Protocol, and Domain Name System (query).

No.	Time	Source	Destination	Protocol	Length	Info
220	6.490031874	34.149.100.209	10.0.2.15	TLSv1.3	1454	Application Data
221	6.490032249	172.217.169.234	10.0.2.15	TLSv1.3	699	Application Data, Application
222	6.490117871	10.0.2.15	34.149.100.209	TCP	54	58826 → 443 [ACK] Seq=1406 Ack
223	6.490189381	10.0.2.15	172.217.169.234	TCP	54	52314 → 443 [ACK] Seq=1362 Ack
224	6.490807403	10.0.2.15	172.217.169.234	TLSv1.3	85	Application Data
225	6.492126957	172.217.169.234	10.0.2.15	TCP	60	443 → 52314 [ACK] Seq=5140 Ack
226	6.496240652	34.149.100.209	10.0.2.15	TLSv1.3	4434	Application Data, Application
227	6.496330777	10.0.2.15	34.149.100.209	TCP	54	58826 → 443 [ACK] Seq=1406 Ack
228	6.498071487	34.149.100.209	10.0.2.15	TLSv1.3	5894	Application Data, Application
229	6.498071682	34.149.100.209	10.0.2.15	TLSv1.3	4434	Application Data, Application
230	6.498851615	10.0.2.15	34.149.100.209	TCP	54	58826 → 443 [ACK] Seq=1406 Ack
231	6.498944932	10.0.2.15	34.149.100.209	TCP	54	58826 → 443 [ACK] Seq=1406 Ack
232	6.500844355	34.149.100.209	10.0.2.15	TLSv1.3	4165	Application Data, Application
233	6.501753127	10.0.2.15	34.149.100.209	TCP	54	58826 → 443 [ACK] Seq=1406 Ack
234	6.501875548	10.0.2.15	34.149.100.209	TLSv1.3	93	Application Data
235	6.502283214	34.149.100.209	10.0.2.15	TCP	60	443 → 58826 [ACK] Seq=38084 Ac
236	6.580750788	34.149.100.209	10.0.2.15	TLSv1.2	2974	Server Hello, Certificate
237	6.580785696	10.0.2.15	34.149.100.209	TCP	54	58868 → 443 [ACK] Seq=219 Ack=
238	6.581767624	34.149.100.209	10.0.2.15	TLSv1.2	211	Server Key Exchange, Server He
239	6.581784475	10.0.2.15	34.149.100.209	TCP	54	58868 → 443 [ACK] Seq=219 Ack=

Frame 1: 88 bytes on wire (704 bits), 88 bytes captured  
Ethernet II, Src: PCSSystemtec ce:db:cf (08:00:27:ce:d  
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 172.  
User Datagram Protocol, Src Port: 59375, Dst Port: 53  
Domain Name System (query)

## Summary:

### IP Addresses:

Source IP addresses: 34.149.100.209, 10.0.2.15, 172.217.169.234

Destination IP addresses: 10.0.2.15, 34.149.100.209, 172.217.169.234

### Protocols Observed:

TLSv1.2 and TLSv1.3 (Transport Layer Security)

TCP (Transmission Control Protocol)

### Packet Information:

Packet lengths vary from 54 bytes to 4434 bytes.

The traffic includes application data, server responses (e.g., "Server Hello, Certificate"), and key exchange information.

### Application-level Details:

The network traffic appears to be related to various internet applications and protocols, including application data transfers, session ticket exchanges, and client key exchanges.

## Task 4: Password Security and Hashing

### Script:

```
import hashlib

password = "mypassword"

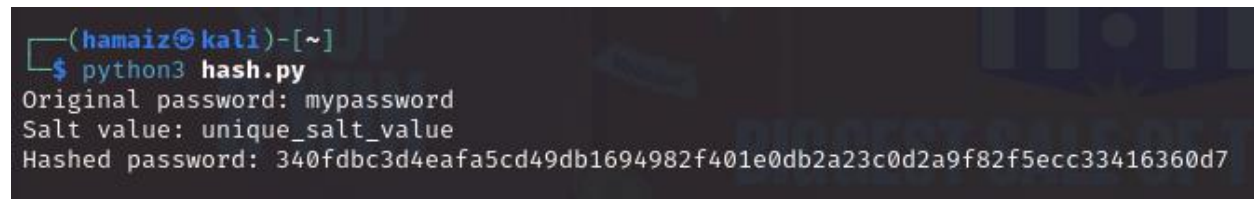
salt = "unique_salt_value"

hashed_password = hashlib.sha256((password + salt).encode()).hexdigest()

print("Original password:", password)

print("Salt value:", salt)

print("Hashed password:", hashed_password)
```



```
(hamaiz@kali)-[~]
$ python3 hash.py
Original password: mypassword
Salt value: unique_salt_value
Hashed password: 340fdb3d4eafa5cd49db1694982f401e0db2a23c0d2a9f82f5ecc33416360d7
```

Hashing is the process of converting data (like a password) into a fixed-length, unique output (the "hash") using a mathematical algorithm. It is a one-way process, meaning that it's computationally infeasible to reverse the hash and retrieve the original password.

Salting is the process of adding a unique, random value (the "salt") to the password before hashing it. It helps mitigate against common attacks like rainbow table attacks, which use precomputed hashes to crack passwords.

### Importance:

**Password Protection:** Hashing and salting are essential for securely storing passwords. Hashed and salted passwords are much harder to crack than plain-text passwords.

**Increased Security:** Salting prevents attacks using pre-computed hash tables, and unique salts make it harder to crack the same password.

**Mitigating Attacks:** Hashing and salting significantly increase the time and resources required for brute-force and dictionary attacks.

# **Task 5: Basic Threat Identification**

## **Common Security Threats in Web Applications**

### **Broken Access Control**

Summary: Broken access control occurs when an application does not properly restrict user access to resources. This can allow unauthorized users to access sensitive data or perform actions they should not be able to.

Example: An attacker might manipulate a URL to access an admin panel by changing their user role in the request, gaining unauthorized access to sensitive administrative functions and data [1].

### **Cryptographic Failures**

Summary: This threat involves the improper implementation of cryptographic algorithms, leading to the exposure of sensitive data. It includes issues like weak encryption, improper key management, and failure to encrypt sensitive data.

Example: An attacker could exploit a web application that uses outdated encryption algorithms (like MD5) to decrypt sensitive user information, such as passwords or credit card numbers, leading to data breaches [2].

### **Injection Attacks**

Summary: Injection attacks occur when an attacker sends untrusted data to an interpreter as part of a command or query. This can allow attackers to execute arbitrary commands or access data they should not be able to.

Example: In a SQL injection attack, an attacker might input a malicious SQL statement into a login form, allowing them to bypass authentication and retrieve user data from the database [3].

### **Security Misconfiguration**

Summary: Security misconfiguration refers to improper settings in an application or server that can lead to vulnerabilities. This can include default settings, incomplete setups, or overly permissive permissions.

Example: An attacker could exploit a web application that has default credentials still in use, allowing them to log in and gain administrative access without any effort [1].



## **Insufficient Logging and Monitoring**

Summary: Insufficient logging and monitoring can prevent organizations from detecting and responding to security incidents effectively. Without proper logs, it becomes difficult to trace unauthorized access or data breaches.

Example: If an application does not log failed login attempts, an attacker could perform a brute-force attack without being detected, potentially compromising user accounts over time [2].

## **Conclusion**

Understanding these common security threats is crucial for developers and organizations to implement effective security measures. By addressing these vulnerabilities, they can significantly reduce the risk of exploitation and protect sensitive data.