

Ethical Hacking Lab

Module 3: Malware Threats: Worms, viruses, Trojans

1. Password Cracking

Aim: To understand the process of generating MD5 hashes for given passwords and using online tools to reverse the hashes, demonstrating basic password cracking techniques

Theory:

MD5 (Message-Digest Algorithm 5):

MD5 is a cryptographic hash function designed by Ronald Rivest in 1991. It takes an input of arbitrary length and produces a fixed 128-bit (16-byte) hash value, typically represented as a 32-character hexadecimal number. MD5 is deterministic, meaning the same input will always produce the same hash output. It is widely used for checksums to verify data integrity, but not for secure hashing due to its vulnerabilities.

- **Characteristics of MD5:**

- Fixed-Length Output: Always produces a 128-bit hash regardless of the input size.
- Fast Computation: Optimized for speed, making it suitable for non-security applications.
- Vulnerabilities: Prone to hash collisions (two different inputs producing the same hash) and easy to crack using modern tools and techniques.

Password Cracking:

Password cracking involves reversing the hashing process or exploiting weaknesses in password storage mechanisms to retrieve plaintext passwords. Modern password systems hash passwords to store them securely, making the hash the target of cracking methods.

Techniques Used in Password Cracking:

1. Brute Force:

Testing all possible combinations of characters to find the password. Time-consuming and computationally expensive for strong passwords.

2. Dictionary Attack:

Using a list of common words or precomputed hashes of common passwords to match the hash.

3. Rainbow Tables:

Precomputed tables of hashes for all possible passwords up to a certain length. Tools like CrackStation utilize these to reverse MD5 hashes efficiently.

4. Salting:

A countermeasure to prevent precomputed attacks by adding random data (salt) to the input before hashing. MD5 without salting is especially vulnerable.

Output:

The screenshot shows a web browser window with the URL md5hashgenerator.com. The page has a header with tabs for "Lab Assignment 3 :Malware thr" and "50-2023 Siddhi Kotre - Lab Ass". The main content area has a "Dan's Tools" logo and a navigation bar with links for "Web Dev", "Conversion", "Encoders / Decoders", "Formatters", "Internet", and "English".

The central part of the page contains a form with the placeholder text "Use this generator to create an MD5 hash of a string:" followed by a text input field containing "Admin12345". Below the input field is a blue "Generate →" button.

A table below the input field displays the results:

Your String	Admin12345
MD5 Hash	e66055e8e308770492a44bf16e875127
SHA1 Hash	459ff8ddc3d877b86573aa391746824c9c1d5c9a

On the left side of the page, there is a vertical banner with the text "Click, Book, Fly!" and a small image of an airplane wing against a blue sky. On the right side, there is another vertical banner with the KLM logo and the text "Travel Well".

At the bottom of the page, a small note reads: "This MD5 hash generator is useful for encoding passwords, credit card numbers and other sensitive data into".

Use this generator to create an MD5 hash of a string:

Ethical@#&Hacking

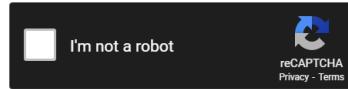
Generate →

Your String	Ethical@#&Hacking
MD5 Hash	d24d22737f3c985997b8715e66e73a8d Copy
SHA1 Hash	bc0fb2f184b8ca1e8ebe8699350f06ac99bbcecc Copy

Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

e66055e8e308770492a44bf16e875127

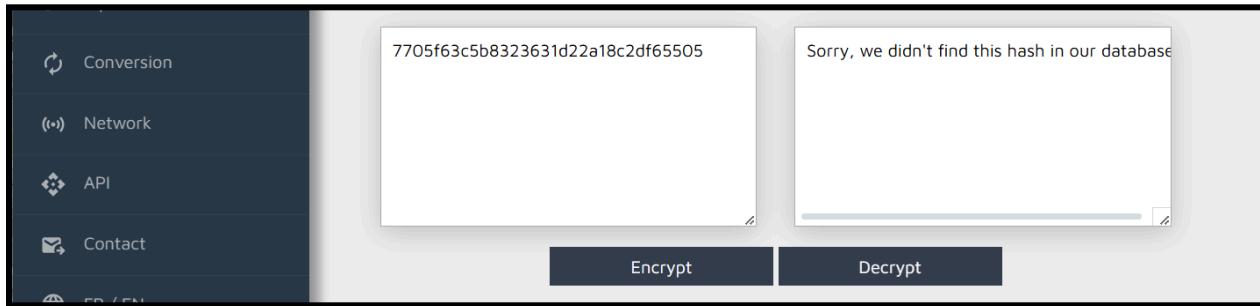
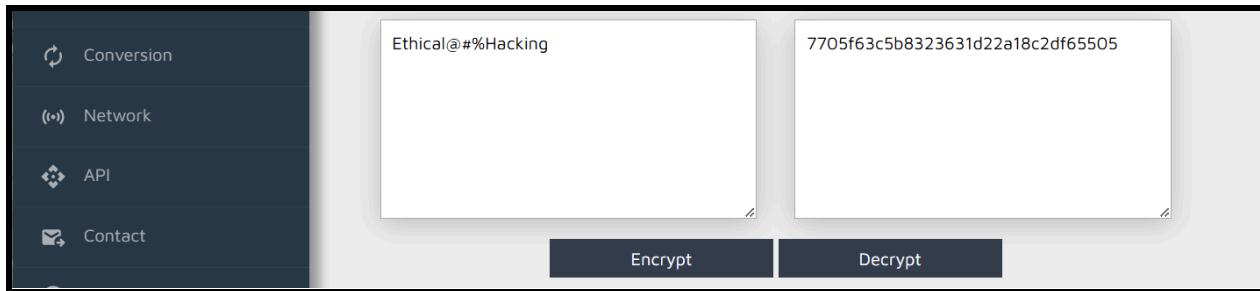


Crack Hashes

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sh1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
e66055e8e308770492a44bf16e875127	md5	Admin12345

Color Codes: Green Exact match, Yellow Partial match, Red Not found.



2. Dictionary Attack

Aim: To perform a dictionary attack on a given MD5 hash using a Python script and a predefined list of possible passwords

Theory:

Dictionary Attack Overview:

A dictionary attack is a method of password cracking where a predefined list of potential passwords (the dictionary) is used to compare hashed outputs with the target hash. Unlike brute-force attacks, which test all possible combinations of characters, a dictionary attack leverages a curated list of likely passwords to increase efficiency.

Steps in a Dictionary Attack:

1. Hash Calculation:

For each word in the dictionary, compute its MD5 hash.

2. Hash Comparison:

Compare the computed hash with the target hash. If a match is found, the password corresponding to the hash is retrieved.

3. Output the Result:

If no match is found after exhausting the dictionary, the attack is unsuccessful.

Code:

dictionary.py

```
import hashlib

def dictionary_attack(md5_hash, passlist_file):
    try:
        with open(passlist_file, 'r') as file:
            for password in file:
                password = password.strip()
                hash_object = hashlib.md5(password.encode('utf-8'))
                hashed_password = hash_object.hexdigest()
                if hashed_password == md5_hash:
                    print(f"[+] Password found: {password}")
                    return
```

```
        print("[-] Password not found in the list.")

except FileNotFoundError:
    print("Error: Password list file not found.")

if __name__ == "__main__":
    md5_hash = input("Enter the MD5 hash value: ")
    passlist_file = input("Enter the password list file name (e.g., passlist.txt): ")
    dictionary_attack(md5_hash, passlist_file)
```

Output:

The screenshot shows a terminal window with the following content:

```
dictionary.py  passlist.txt  passlist.txt
passlist.txt
1 Admin12345
2 hello
3 bye
4 12345
5 siddhi
6 sunshine
7 daisy
8 e66055e8e308770492a44bf16e875127

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

C:\Projects>python dictionary.py
Enter the MD5 hash value: e66055e8e308770492a44bf16e875127
Enter the password list file name (e.g., passlist.txt): passlist.txt
[+] Password found: Admin12345

C:\Projects>[]
```

The screenshot shows a terminal window with the following content:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

C:\Projects>python dictionary.py
Enter the MD5 hash value: e66055e8e308770492a44bf16e875127
Enter the password list file name (e.g., passlist.txt): passlist.txt
[+] Password found: Admin12345

C:\Projects>[]
```

3. Encrypt and Decrypt Passwords

Aim: To demonstrate how to encrypt and decrypt a password using different encryption algorithms and tools

Theory:

Encryption is the process of converting data (in this case, a password) into a format that is unreadable without the proper decryption key or algorithm. This is done to protect sensitive information.

Why Encrypt and Decrypt Passwords?

While hashing is the recommended practice for password storage in most scenarios, encryption can be useful in situations where:

1. The application needs reversible access to the plaintext password (e.g., certain legacy systems).
2. Sensitive data beyond passwords, such as API keys or user credentials, requires protection.

When dealing with passwords, **encryption** and **decryption** differ fundamentally from **hashing**. Hashing is one-way (irreversible), while encryption is two-way, allowing retrieval of the original data with the correct decryption key.

How It Works:

1. Key Generation:

- The `generate_key()` function creates a unique encryption key and saves it to a file (default: `key.key`).
- This key is crucial for both encryption and decryption.

2. Encryption:

- The `encrypt_password()` function takes a plaintext password and encrypts it using the key.
- The result is an unreadable encrypted string.

3. Decryption:

- The `decrypt_password()` function takes the encrypted password and decrypts it back to its original form using the same key.

1. Encrypt Using Encode-Decode

Output:

A screenshot of a web browser showing the encode-decode.com/encryption-functions/ page. The page title is "encrypt & decrypt online". On the left, there is a sidebar with social media sharing icons for Facebook, Twitter, Google+, Pinterest, and LinkedIn. The main input field contains the text "Admin12345". To the right of the input field, the encrypted output is displayed as "DrbqRU8XKe0W8TmehnnnmLw==". Below the input field is a text area labeled "Paste secret." which is currently empty. At the bottom, there are two buttons: a blue "Encrypt string →" button on the left and a green "← Decrypt string" button on the right. A dropdown menu at the top right indicates "supported encryptions: aes-128-cbc".

A screenshot of a web browser showing the "idea encrypt & decrypt online" page. The page title is "idea encrypt & decrypt online". Similar to the previous screenshot, it features social media sharing icons on the left. The main input field contains the text "Admin12345". To the right of the input field, the encrypted output is displayed as "1Ffd0BERMASgXul8WcrQQQ==". Below the input field is a text area labeled "secret" which is currently empty. At the bottom, there is a single green "← Decrypt string" button. A dropdown menu at the top right indicates "supported encryptions: idea".

2. Encrypt and Decrypt Using OpenSSL

Why Use OpenSSL for Encryption and Decryption?

- OpenSSL provides command-line tools and libraries for encryption tasks.
- It is secure, efficient, and supports a wide range of cryptographic algorithms.
- It is widely used in TLS/SSL for secure web communications.

Output:

```
siddhikotre@siddhikotre:~$ sudo apt install openssl
[sudo] password for siddhikotre:
Sorry, try again.
[sudo] password for siddhikotre:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
openssl is already the newest version (3.0.2-0ubuntu1.18).
openssl set to manually installed.
The following packages were automatically installed and are no longer required:
  libwpe-1.0-1 libwpebackend-fdo-1.0-1
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 67 not upgraded.
siddhikotre@siddhikotre:~$ openssl version
OpenSSL 3.0.2 15 Mar 2022 (Library: OpenSSL 3.0.2 15 Mar 2022)
siddhikotre@siddhikotre:~$
```

```
siddhikotre@siddhikotre:~$ echo "Admin12345" | openssl enc -aes128 -a -salt -pbkdf2 -pass pass:mysecurepassword
U2FsdGVkX1/MCL8z1mxnTDhQwiSh6mD4AxW4tP+uQS4=
siddhikotre@siddhikotre:~$ ^C
siddhikotre@siddhikotre:~$ echo "U2FsdGVkX1/MCL8z1mxnTDhQwiSh6mD4AxW4tP+uQS4=" | openssl enc -aes128 -a -d -salt -pbkdf2 -pass pass:mysecurepassword
Admin12345
siddhikotre@siddhikotre:~$
```

4. DoS Attack

Aim: To understand and explain different types of Denial of Service (DoS) attacks, specifically focusing on:

1. The Ping of Death
2. TCP SYN Flooding
3. The Smurf Attack

Theory:

Description:

The Ping of Death is a type of DoS attack that exploits vulnerabilities in older systems by sending malformed or oversized packets using the Internet Control Message Protocol (ICMP), typically via the "ping" command.

How It Works:

1. Normally, the ICMP protocol limits packet sizes to 65,535 bytes.
2. The attacker sends packets larger than this limit, often fragmented into smaller pieces.
3. The target system reassembles the oversized packet, exceeding buffer limits and causing crashes, reboots, or instability.

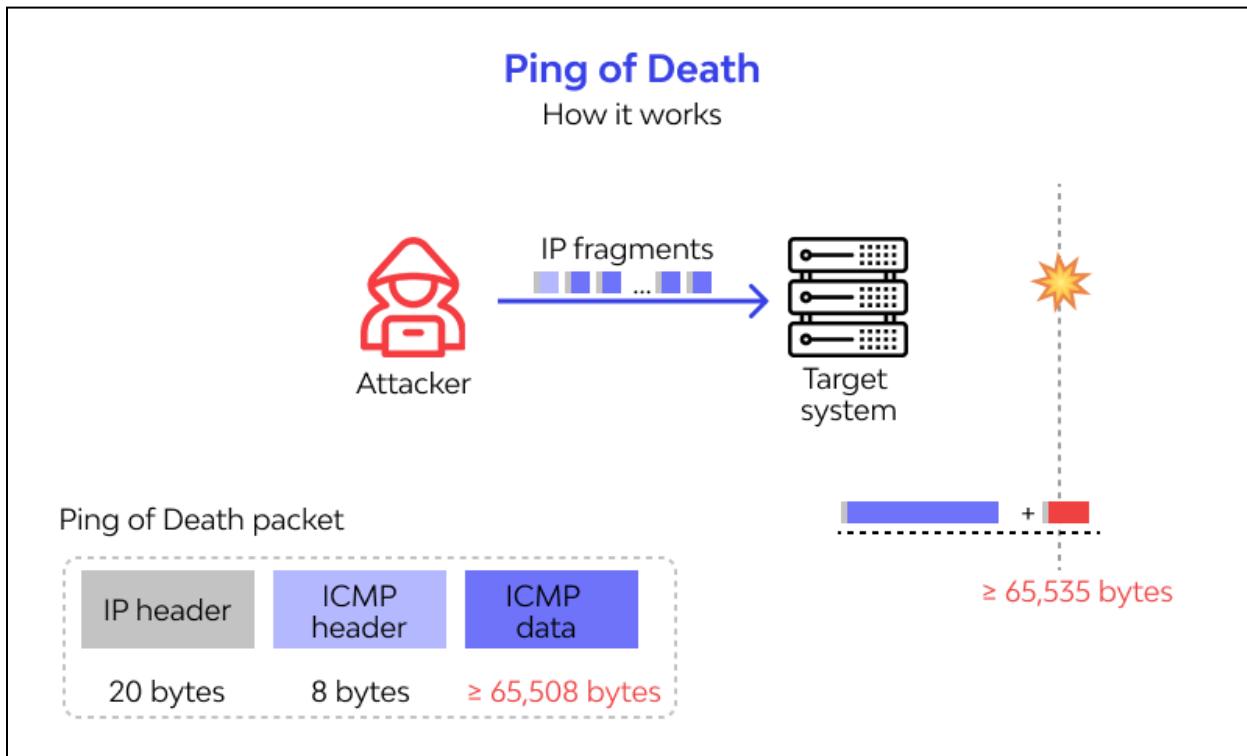
Impact:

- System crashes or freezes.
- Reboots or loss of connectivity.

Modern Mitigations:

- Updated operating systems and network devices now handle oversized packets correctly.
- Firewalls block unusually large ICMP packets.

Diagram:



Theory: TCP SYN Flooding

Description:

TCP SYN Flooding is a DoS attack that exploits the **TCP three-way handshake** process. It overwhelms the server by sending a flood of TCP SYN requests, leaving resources tied up in half-open connections.

How It Works:

1. The attacker sends a large number of TCP SYN packets (connection requests) to the target server.
2. The server allocates resources and responds with SYN-ACK packets, waiting for the attacker's ACK to complete the handshake.
3. The attacker does not respond with an ACK, leaving the connection in a "half-open" state.
4. Legitimate users cannot connect because the server's resources are exhausted.

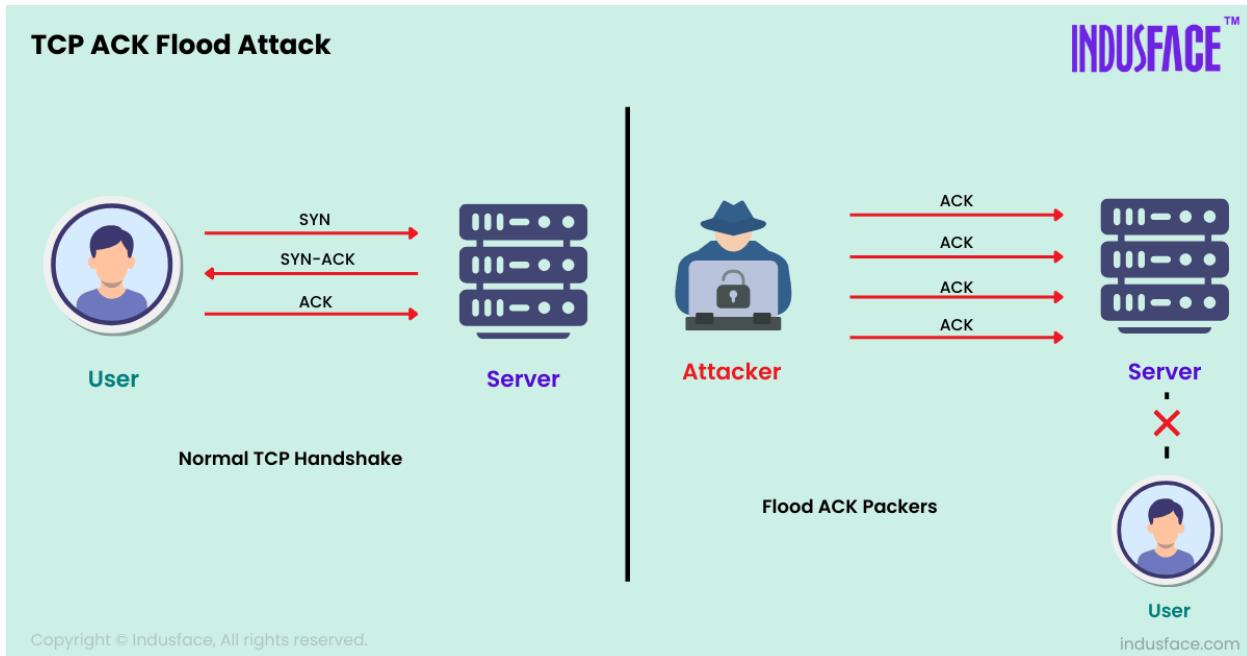
Impact:

- Exhaustion of server memory and processing resources.
- Legitimate users unable to establish connections.

Modern Mitigations:

- **SYN Cookies:** A technique that avoids allocating resources for SYN requests until the handshake is complete.
- **Rate Limiting:** Limits the number of SYN requests from a single IP address.
- **Firewalls and Intrusion Prevention Systems (IPS):** Detect and block SYN flooding attempts.

Diagram:



Theory: The Smurf Attack

Description:

The Smurf Attack is a type of **amplification attack** that exploits ICMP and broadcast addresses. It amplifies traffic sent to the target by tricking other systems into flooding the victim with responses.

How It Works:

1. The attacker sends ICMP Echo Request (ping) packets to a network broadcast address, with the source IP address spoofed to match the victim's IP.
2. All devices in the network respond with ICMP Echo Replies, directing the traffic to the victim.
3. This floods the victim with massive amounts of traffic.

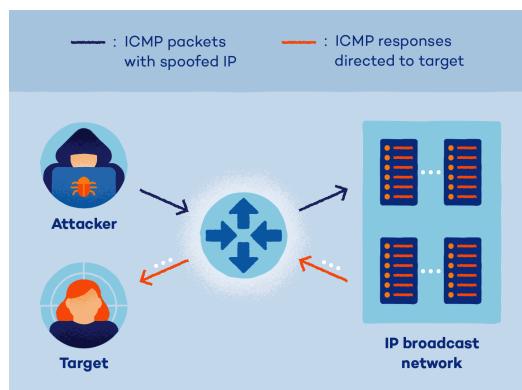
Impact:

- Overwhelms the victim's network with traffic.
- Legitimate users cannot access the victim's resources.

Modern Mitigations:

- Configure routers to block directed broadcast traffic.
- Use firewalls to filter out spoofed packets.
- Implement network ingress and egress filtering to prevent IP spoofing.

Diagram:



5. ARP Poisoning in Windows

Aim: To understand and demonstrate ARP poisoning in a Windows environment using the tool Cain & Abel, with a focus on intercepting and analyzing network traffic

Theory:

ARP Poisoning, also known as **ARP Spoofing**, is a type of attack where an attacker sends malicious ARP (Address Resolution Protocol) packets onto a local network. These packets aim to associate the attacker's MAC address with the IP address of another device on the network, such as a gateway or a victim's device. This allows the attacker to intercept, modify, or disrupt communication between devices.

How ARP Poisoning Works

1. Normal ARP Functionality:

- Devices on a local network use ARP to resolve IP addresses to MAC addresses.
- For example, if a device wants to communicate with **192.168.1.1**, it broadcasts an ARP request asking for the MAC address of that IP.

2. Exploiting ARP Vulnerability:

- ARP is inherently insecure because it lacks authentication.
- Any device can respond to an ARP request, claiming any IP-to-MAC mapping.

3. The Attack Process:

- The attacker sends fake ARP responses to devices on the network, mapping the target's IP (e.g., a router or a victim's device) to the attacker's MAC address.
- The devices update their ARP cache with the attacker's MAC address for the targeted IP.
- Future traffic intended for the target device is sent to the attacker instead.

4. Result:

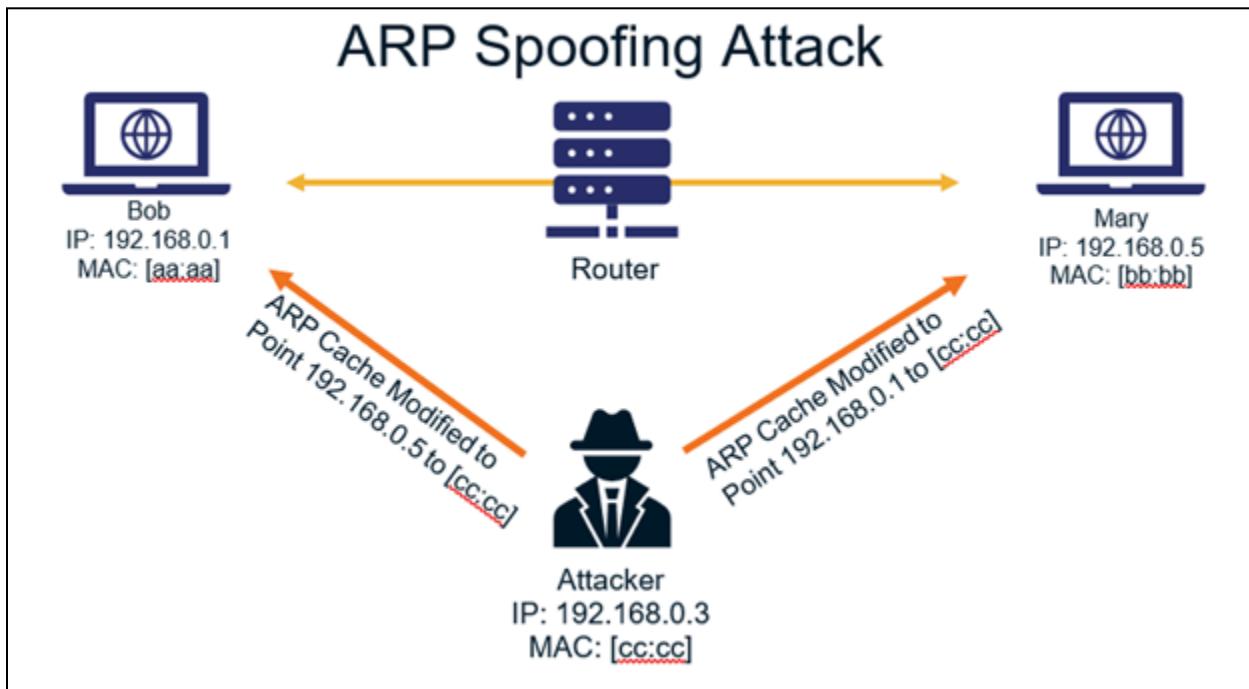
- **Interception:** The attacker can intercept data between devices (Man-in-the-Middle attack).
- **Manipulation:** The attacker can modify the intercepted data.
- **Denial of Service (DoS):** The attacker can drop or disrupt the traffic.

ARP Poisoning on a Windows Network

Signs of ARP Poisoning

- Network communication is unusually slow.
- Duplicate IP address warnings in Windows.
- Unexpected disconnections from the network.
- Suspicious ARP entries in the ARP cache (viewable with `arp -a` command).

Diagram:



6. Ifconfig, Ping, Netstat, Traceroute

Aim: To understand and demonstrate how to use the basic network troubleshooting commands Ipconfig, Ping, Netstat, and Traceroute (Tracert) to gather information and diagnose network issues in a TCP/IP environment

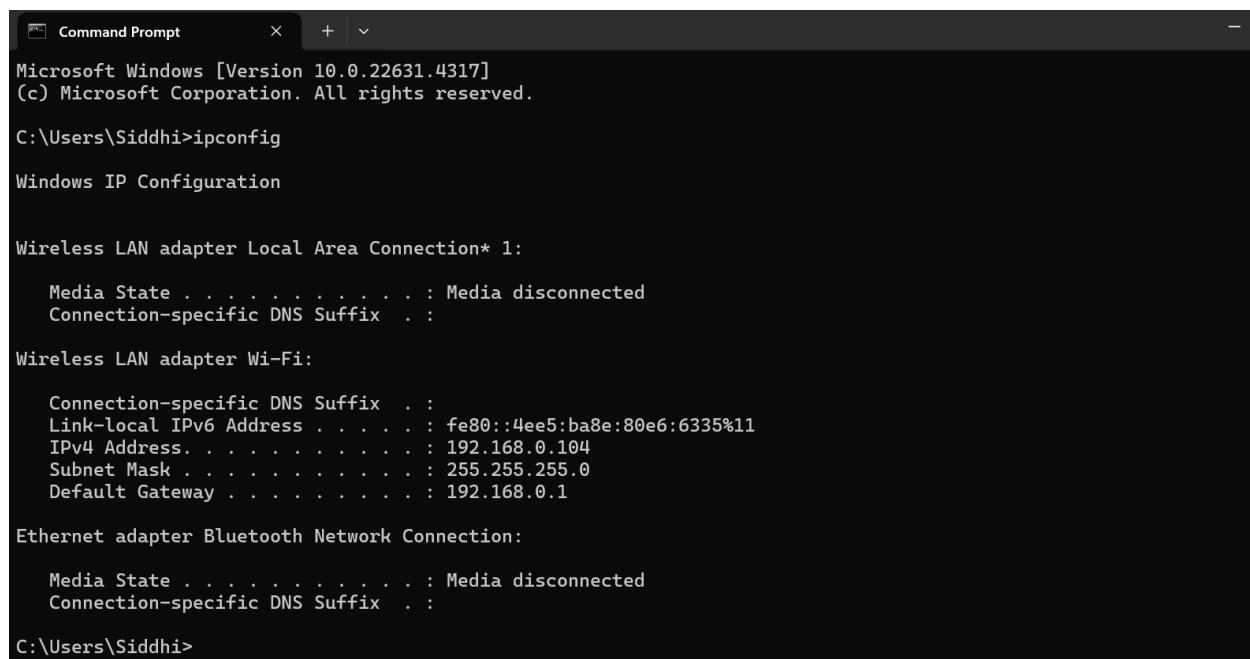
Theory:

Ipconfig is a command-line tool used in Windows to display the current network configuration of a computer. It shows the IP address, subnet mask, and default gateway of network adapters. It is often used to troubleshoot network connectivity issues.

Key Concepts:

- **IP Address:** A unique identifier assigned to a device on a network.
- **Subnet Mask:** Defines the range of IP addresses within a network.
- **Default Gateway:** The device used to route traffic from your local network to other networks.

Output:



```
Command Prompt + 
Microsoft Windows [Version 10.0.22631.4317]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Siddhi>ipconfig

Windows IP Configuration

Wireless LAN adapter Local Area Connection* 1:
  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . . .

Wireless LAN adapter Wi-Fi:
  Connection-specific DNS Suffix . . .
  Link-local IPv6 Address . . . . . : fe80::4ee5:ba8e:80e6:6335%11
  IPv4 Address . . . . . : 192.168.0.104
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 192.168.0.1

Ethernet adapter Bluetooth Network Connection:
  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . . .

C:\Users\Siddhi>
```

Theory:

Ping is a network utility used to test the reachability of a host or IP address on a network. It sends ICMP Echo Request packets and waits for an Echo Reply. It is useful for checking network connectivity and diagnosing packet loss or latency issues.

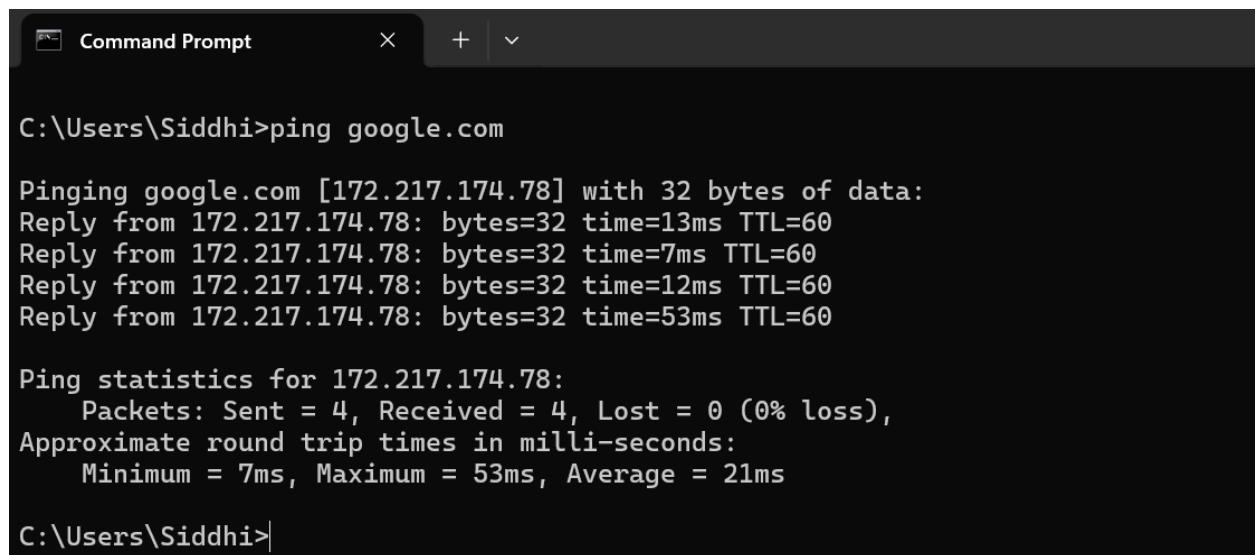
Key Concepts:

- **ICMP:** Internet Control Message Protocol, used for diagnostic purposes.
- **Round-Trip Time (RTT):** The time it takes for a packet to travel from the source to the destination and back.

Steps to be followed:

1. Open **Command Prompt**.
2. Type *ping [destination IP or hostname]* and press Enter. For example, *ping google.com*.
3. Observe the round-trip time (RTT) and packet loss information.
4. If packets are successfully returned, the destination is reachable. If not, there may be a network issue.

Output:



```
Command Prompt + ▾
C:\Users\Siddhi>ping google.com

Pinging google.com [172.217.174.78] with 32 bytes of data:
Reply from 172.217.174.78: bytes=32 time=13ms TTL=60
Reply from 172.217.174.78: bytes=32 time=7ms TTL=60
Reply from 172.217.174.78: bytes=32 time=12ms TTL=60
Reply from 172.217.174.78: bytes=32 time=53ms TTL=60

Ping statistics for 172.217.174.78:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 7ms, Maximum = 53ms, Average = 21ms

C:\Users\Siddhi>
```

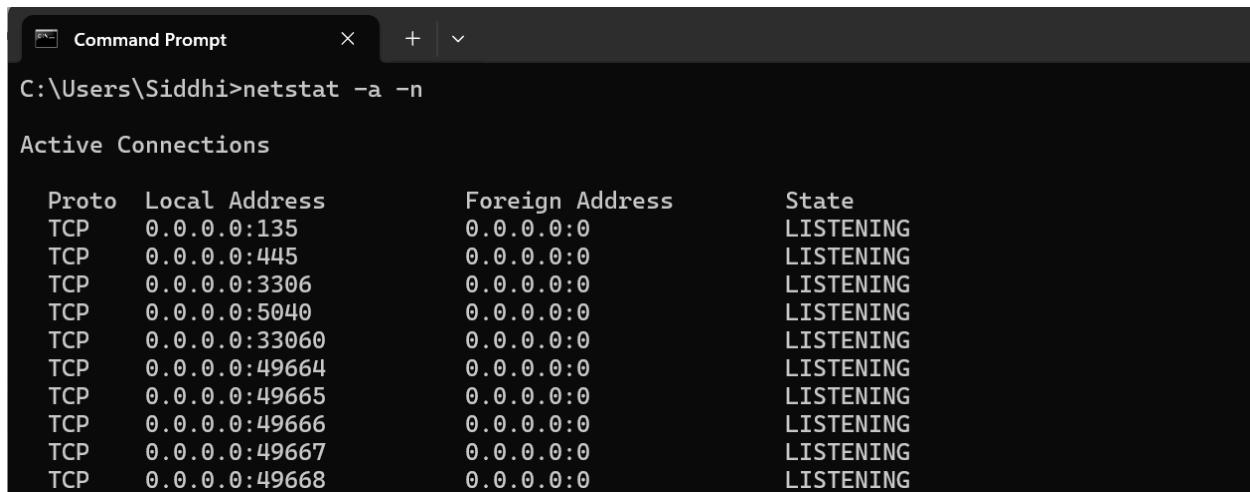
Theory:

Netstat (Network Statistics) is a command-line tool used to display network connections, routing tables, interface statistics, and other network-related information. It can be helpful for identifying active connections, listening ports, and network traffic.

Key Concepts:

- **Active Connections:** Open communication paths between devices on the network.
- **Listening Ports:** Ports on which a system is waiting for incoming connections.

Output:



A screenshot of a Windows Command Prompt window titled "Command Prompt". The window shows the output of the command "netstat -a -n". The output displays "Active Connections" with columns for Proto, Local Address, Foreign Address, and State. All entries show TCP protocols with local addresses ranging from 0.0.0.0:135 to 0.0.0.0:49668 and foreign addresses as 0.0.0.0:0, all in a LISTENING state.

Proto	Local Address	Foreign Address	State
TCP	0.0.0.0:135	0.0.0.0:0	LISTENING
TCP	0.0.0.0:445	0.0.0.0:0	LISTENING
TCP	0.0.0.0:3306	0.0.0.0:0	LISTENING
TCP	0.0.0.0:5040	0.0.0.0:0	LISTENING
TCP	0.0.0.0:33060	0.0.0.0:0	LISTENING
TCP	0.0.0.0:49664	0.0.0.0:0	LISTENING
TCP	0.0.0.0:49665	0.0.0.0:0	LISTENING
TCP	0.0.0.0:49666	0.0.0.0:0	LISTENING
TCP	0.0.0.0:49667	0.0.0.0:0	LISTENING
TCP	0.0.0.0:49668	0.0.0.0:0	LISTENING

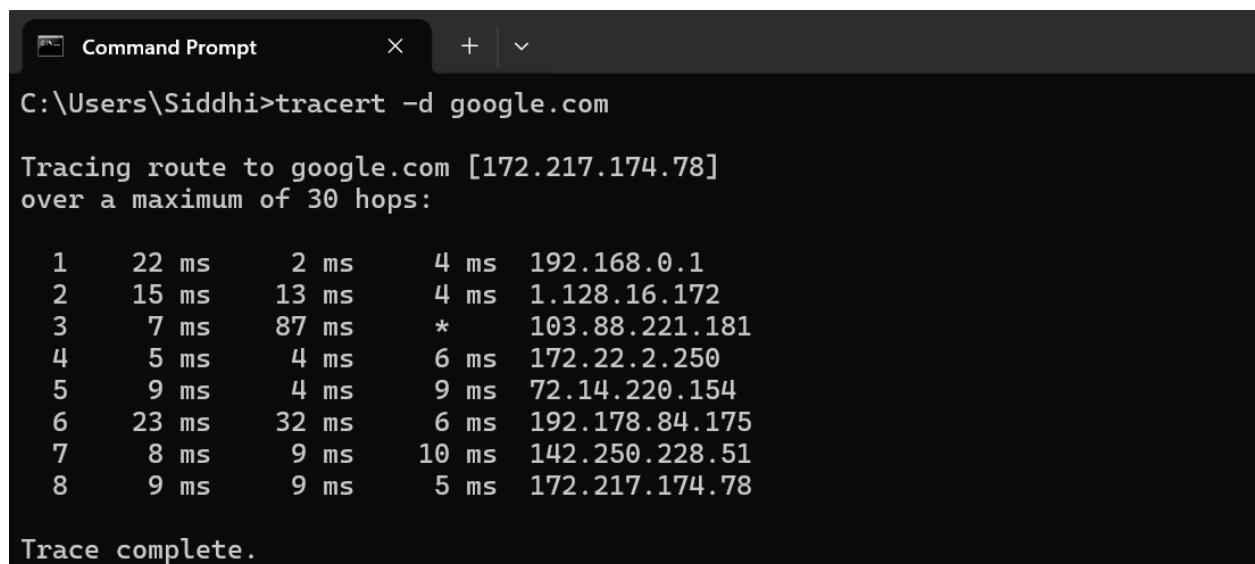
Theory:

Traceroute (or **Tracert** on Windows) is a command used to trace the path that packets take from one device to another across a network. It helps identify routing issues by displaying each hop along the way and the time it takes to reach each hop.

Key Concepts:

- **Hops:** Intermediate devices (routers) that packets pass through to reach their destination.
- **TTL (Time-to-Live):** A value that limits the number of hops a packet can take before being discarded.

Output:



```
Command Prompt
C:\Users\Siddhi>tracert -d google.com

Tracing route to google.com [172.217.174.78]
over a maximum of 30 hops:

 1  22 ms      2 ms      4 ms  192.168.0.1
 2  15 ms      13 ms      4 ms  1.128.16.172
 3  7 ms       87 ms     *      103.88.221.181
 4  5 ms       4 ms      6 ms  172.22.2.250
 5  9 ms       4 ms      9 ms  72.14.220.154
 6  23 ms      32 ms      6 ms  192.178.84.175
 7  8 ms       9 ms     10 ms  142.250.228.51
 8  9 ms       9 ms      5 ms  172.217.174.78

Trace complete.
```

7. Steganography Tools

Aim: To understand and demonstrate the process of Steganography, specifically using a Steganography tool to hide data in images (such as BMP files) and retrieve it effectively

Theory:

Steganography is the practice of concealing information within other non-suspicious data, such as images, audio, or video files. Unlike encryption, which alters the data to make it unreadable without a key, steganography hides the data in plain sight, often making it invisible to the human eye or ears.

Types of Steganography:

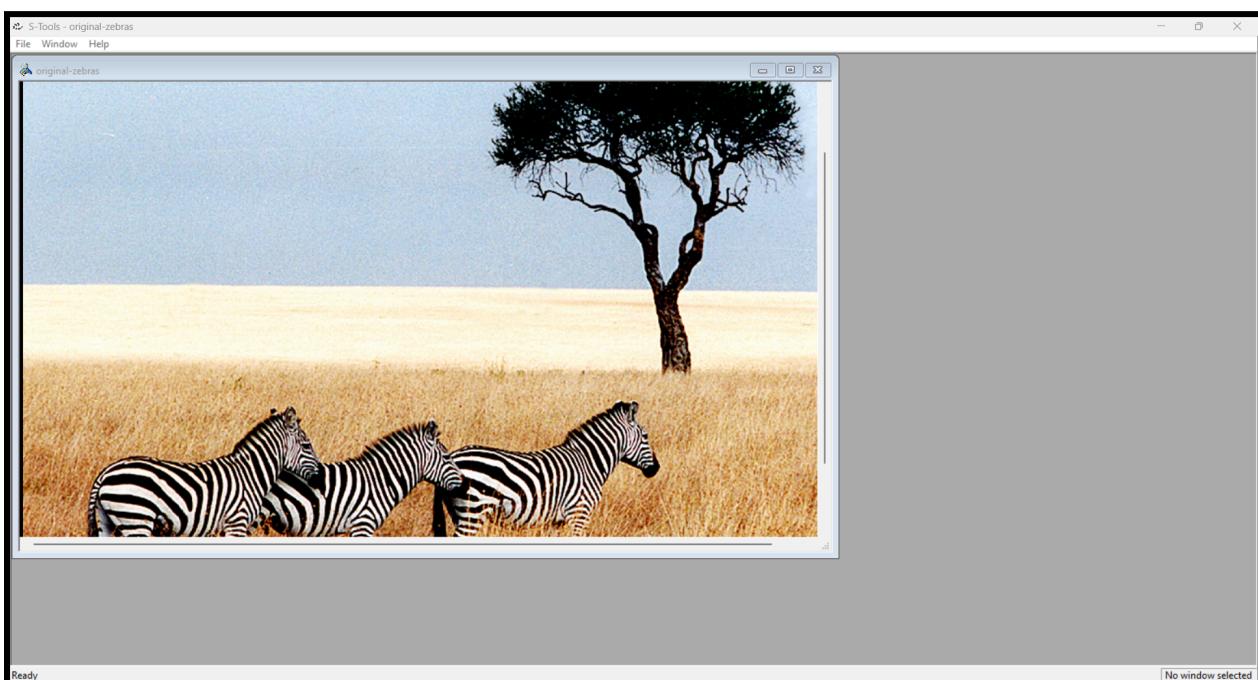
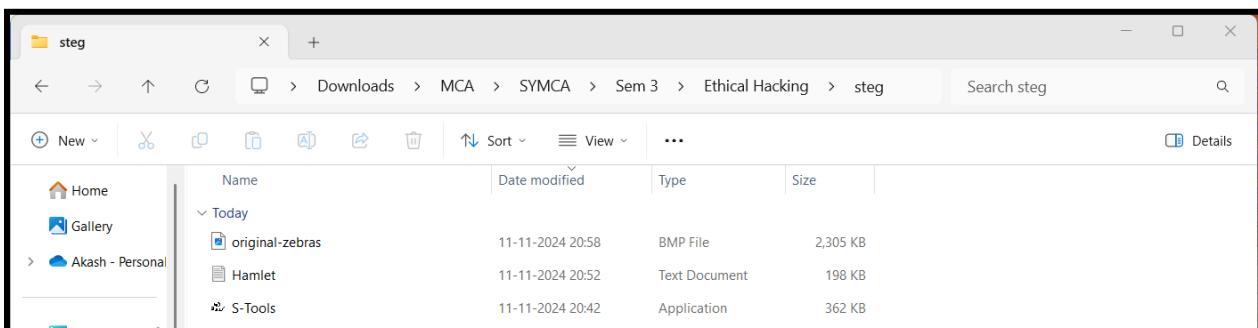
- **Image Steganography:** Hides information in images by manipulating pixel data, often in the least significant bits (LSB) of the image.
- **Audio Steganography:** Conceals data in audio files by modifying the audio signal in a way that is imperceptible to the human ear.
- **Text Steganography:** Embeds hidden information within text, using techniques such as altering spaces, punctuation, or word choices.

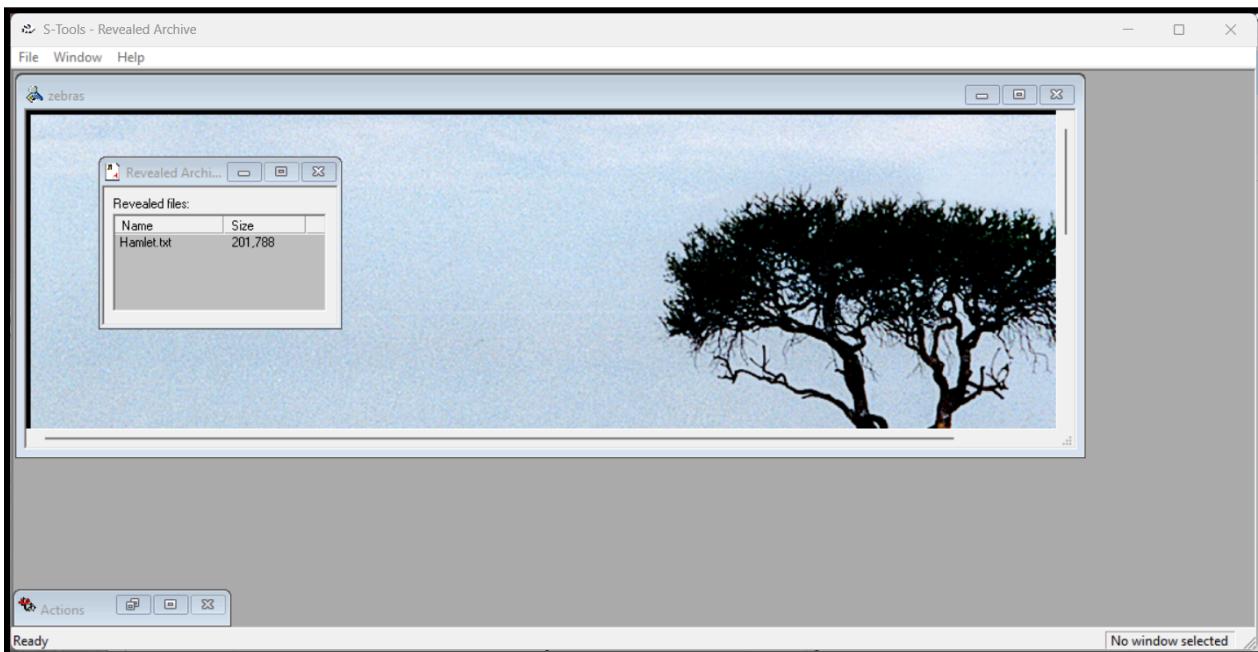
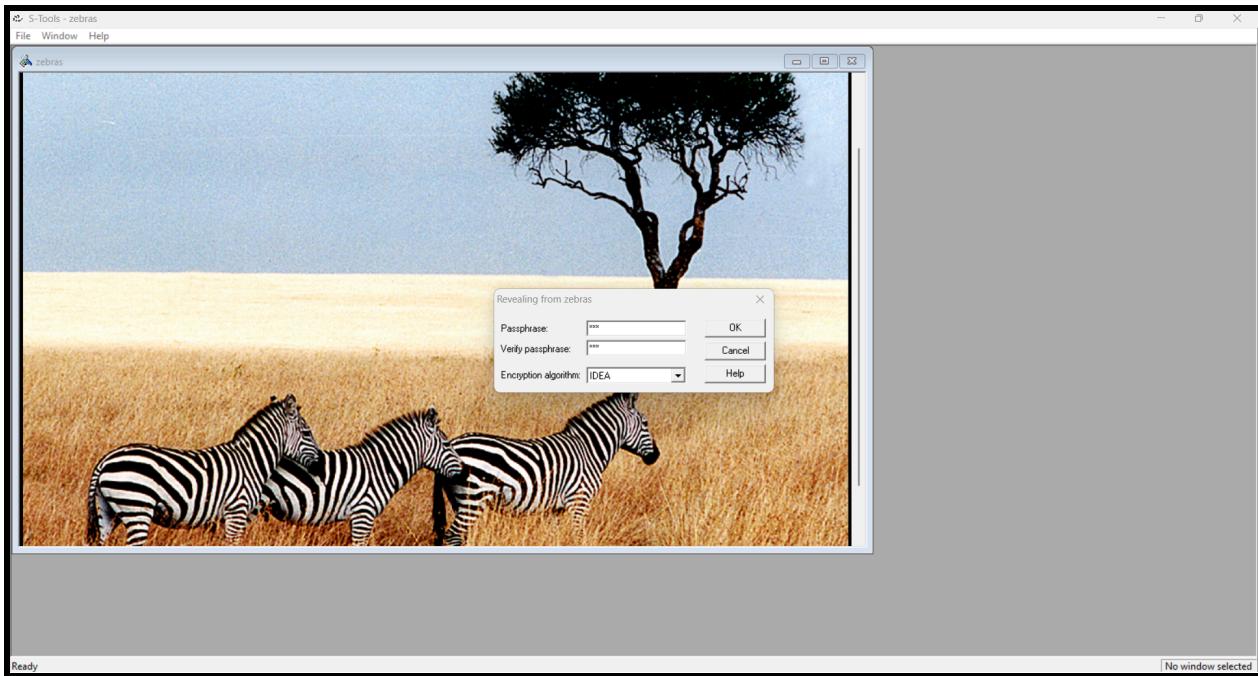
Steganography in BMP Files

BMP (Bitmap) files are commonly used for image-based steganography due to their uncompressed nature, making them easier to manipulate at the pixel level. The data is typically hidden in the least significant bits (LSB) of the image pixels, which are barely noticeable when viewed by the human eye.

- **Least Significant Bit (LSB) Encoding:** This method modifies the least significant bit of each pixel in an image. Since these changes are subtle, they are typically not detectable by the human eye but can be decoded to reveal hidden information.

Output:





The screenshot shows a text editor window titled "Hamlet.txt". The content is a scanned copy of the first few pages of the play. It includes the year "1604", the title "THE TRAGEDY OF HAMLET, PRINCE OF DENMARK", and the author "by William Shakespeare". Below this, the "Dramatis Personae" section lists numerous characters: Claudius, King of Denmark; Marcellus, Officer; Hamlet, son to the former, and nephew to the present king; Polonius, Lord Chamberlain; Horatio, friend to Hamlet; Laertes, son to Polonius; Voltimand, courtier; Cornelius, courtier; Rosencrantz, courtier; Guildenstern, courtier; Osric, courtier; A Gentleman, courtier; A Priest; and others. At the bottom of the window, status bars indicate "Ln 1, Col 1", "1,96,848 characters", "100%", "Windows (CRLF)", and "UTF-8".

1604

THE TRAGEDY OF HAMLET, PRINCE OF DENMARK

by William Shakespeare

Dramatis Personae

Claudius, King of Denmark.
Marcellus, Officer.
Hamlet, son to the former, and nephew to the present king.
Polonius, Lord Chamberlain.
Horatio, friend to Hamlet.
Laertes, son to Polonius.
Voltimand, courtier.
Cornelius, courtier.
Rosencrantz, courtier.
Guildenstern, courtier.
Osric, courtier.
A Gentleman, courtier.
A Priest.
etc.

Ln 1, Col 1 1,96,848 characters 100% Windows (CRLF) UTF-8