### 

**CYBER SECURITY**

**Project Title:Keylogger**

**Group Members:**

**Abdul Haseeb 23k-2029**

**Muzammil 23k-2023**

**Zain 23k-2038**

### ***Submitted to:*** Dr.Fahad Samad

### ****Report on Cybersecurity Project: Keylogger****

## ****Introduction****

This report presents a comprehensive analysis of a simulated Cyber Security project based on keylogger that demonstrates how an attacker could exploit vulnerabilities within a local network to compromise a victim's system. The primary objective of this project is to study the methods used by attackers to deliver and execute a Python-based keylogger and understand the implications of such attacks.

The project scenario revolves around an attacker and a victim sharing the same local network, and it emphasizes key techniques such as **reconnaissance**, **brute-force attacks**, **payload delivery**, and **credential harvesting**. The findings serve to highlight security weaknesses and recommend mitigation strategies to prevent such breaches.

## ****Scenario Overview****

1. **Victim**: An office employee using a laptop connected to a local network.
2. **Attacker**: A malicious entity on the same network.
3. **Goal**: To exploit vulnerabilities in the victim’s system to deliver a keylogger and harvest sensitive credentials.

## Tools and services Used:

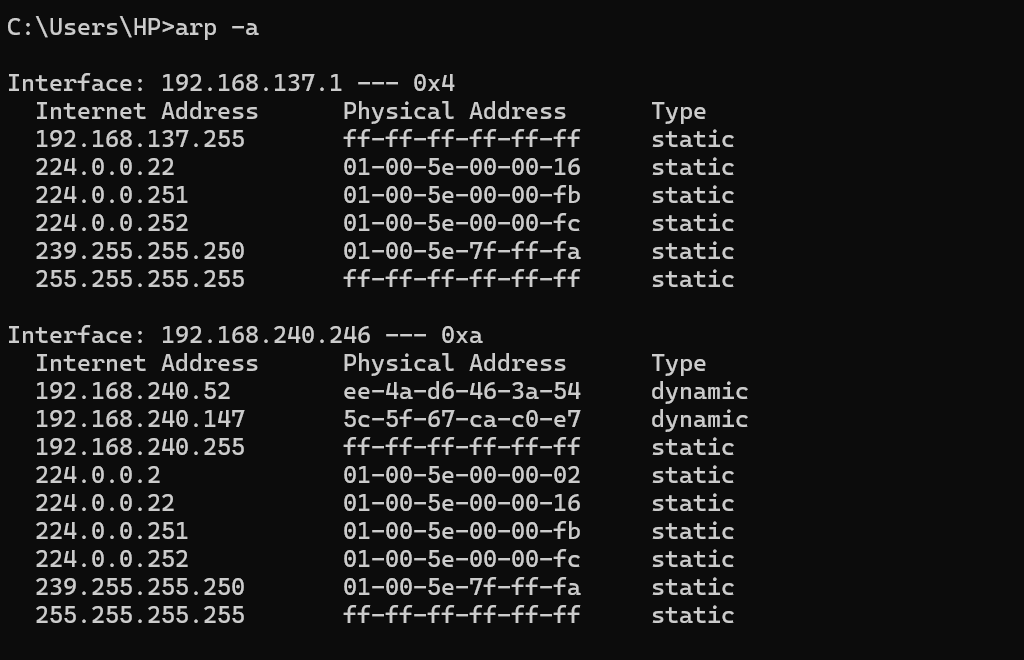
* NMAP
* SSH
* HYDRA
* PYNPUT
* LINUX(WSL) AND WINDOWS

## ****Steps of the Attack****

### ****Step 1: Reconnaissance****

The attacker begins by identifying devices connected to the local network using cmd.exe as:

* arp -a: Lists all devices on the local network along with their IP and MAC addresses.





#### ****Findings****:

The attacker identifies the victim’s IP address ( 192.168.240.147) as it is type:dynamic and the first dynamic ip is the host gateway ip.

### ****Step 2: Scanning for Vulnerabilities****

* **NMAP**: A powerful tool for network scanning and port discovery.

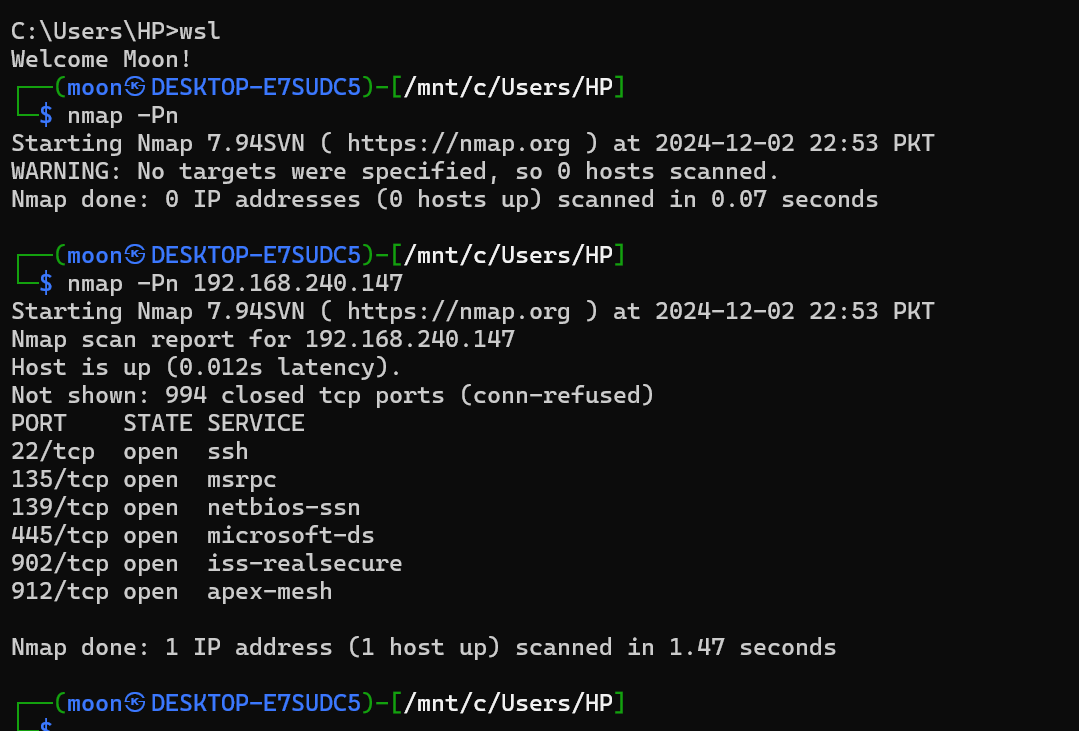
The attacker uses nmap to perform a deeper scan of the victim's device:

nmap -Pn -p 192.168.240.147

-Pn(No ping) flag will mark the target as already up.

This is useful when:Firewalls or intrusion detection systems block ICMP (ping) requests.

The scan using nmap confirms the presence of an SSH service running on Port 22.



#### ****Findings****:

The attacker identifies the victim’s IP address (192.168.240.147) and determines that Port 22 (SSH) is open.

#### ****About SSH****:

Secure Shell (SSH) is a protocol used for secure remote access to a computer over an unsecured network. It enables encrypted communication, commonly used for remote administration. However, if weak passwords are used, attackers can brute-force credentials to gain unauthorized access.

### ****Step 3: Brute-Force Attack on SSH****

After finding this vulnerability in target system,attacker needs to exploit it.The attacker leverages **Hydra**, a powerful brute-force tool, to crack the SSH login credentials. Using the rockyou.txt wordlist, the attacker initiates a brute-force attack:

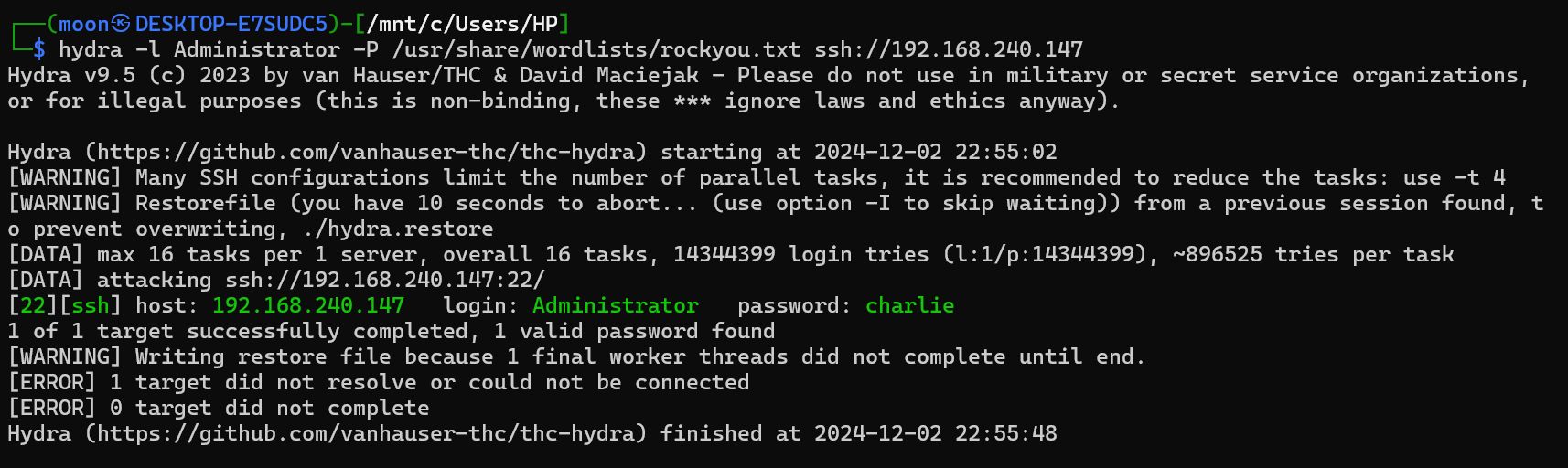
hydra -l username -P /path/to/rockyou.txt ssh://192.168.1.10

Hydra -l Administrator -P /usr/share/wordlists/rockyou.txt ssh://192.168.240.147

#### ****Outcome****:

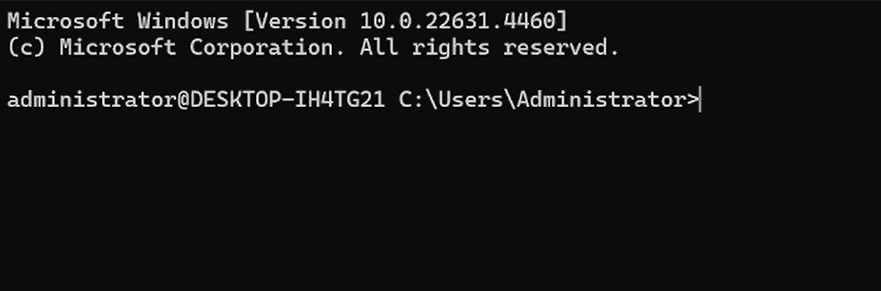
Great,After multiple attempts, the attacker successfully retrieves the victim’s credentials (e.g., Pa:password123) and gains shell access to the victim’s machine.

Here attacker retrieves password: *charlie*



### ****Step 4: Logging into SSH to access shell using credentials.****

### 



Great.Now the attacker has access to target machine.

### ****Step 5: Delievering the Keylogger and installing of necessary dependencies via SSH(remotely).****

With shell access, the attacker uploads a Python script (keylogger.py) to the victim's machine in a directory where it cannot see on its usual workprocess.

#### ****Keylogger Details****

* **Language**: Python
* **Library Used**: pynput for capturing keystrokes.
* **Purpose**: Records all keystrokes entered by the victim and transmits them to the attacker’s system where a listener will record it.
* **Efficiency:** Records keystrokes with exact date and time.
* **Storage**: Keystrokes are saved in logs.txt on the attacker’s machine for later analysis and also will be saved to Desktop of the victim(here saved only for test purpose i.e can be removed).

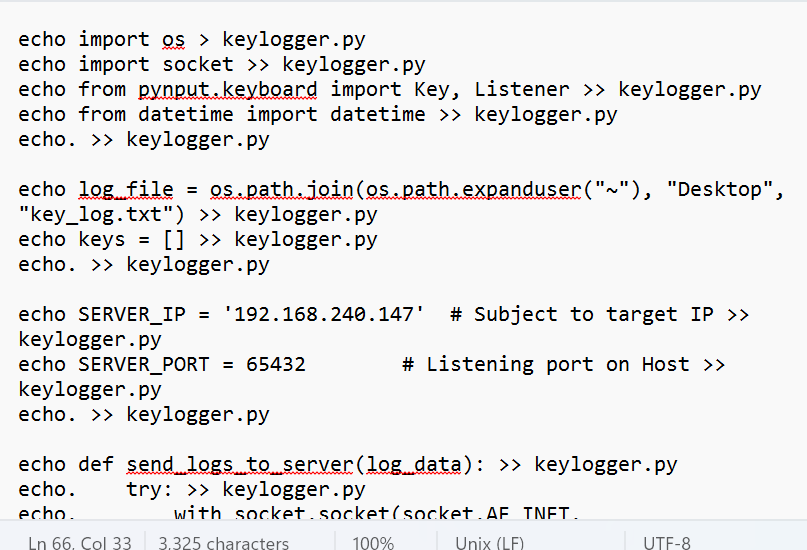
**Description:**

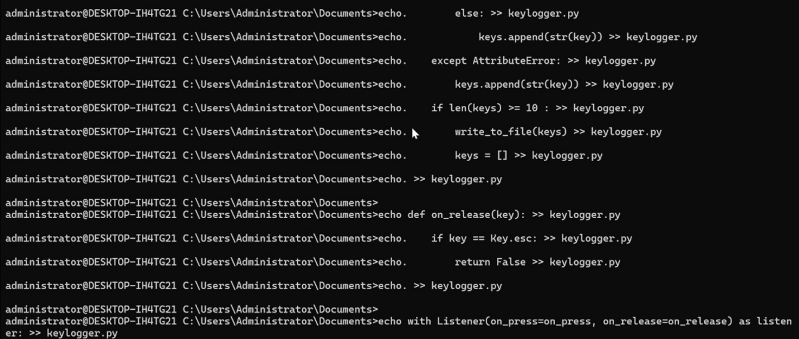
The scripts works on the basis of pynput library which is used to record keystrokes and then sends the strokes to sever which is running on attacker pc using socket programming.The scripts shows its work when it writes the keystrokes to a file named logs.txt;proper handles have been used rto record alphabets,numeric,special characters and space keys.It shows its efficiency when it also saves the logs with the matching date and time and also handles proper errors.

#### ****Keylogger Script****

//saved as keylogger.py

//paste the file named on target shell named as echo\_keylogger.txt.

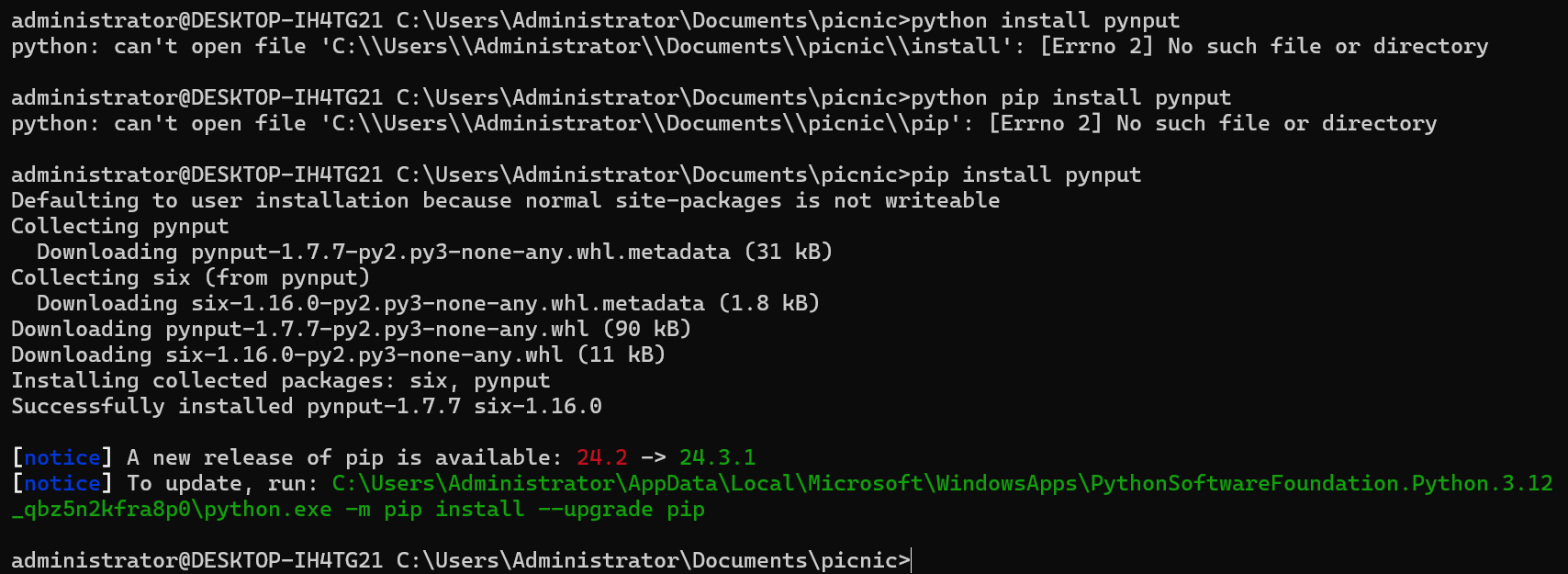




//The attacker cannot open Notepad or any text editor on target PC directly so he has to write every single line on victim PC using the shell.This can be done by using “**echo”**.

**ON CMD.exe:** echo ‘instruction’ >> filename.py

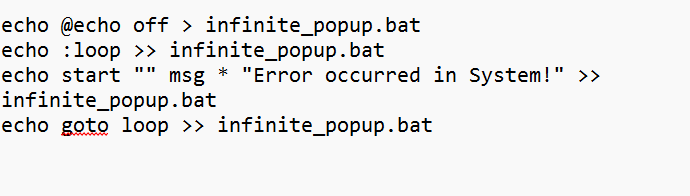
**Installing of necessary dependencies:(pynput)**



### ****Step 6: Victim Disruption****

To prevent the victim from interfering during the attack, the attacker creates a malicious batch script (infinite\_error.bat) on the victim’s machine:

//saved as infinite\_error.bat



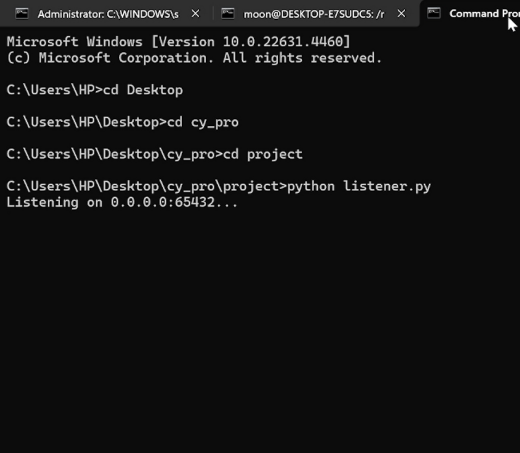
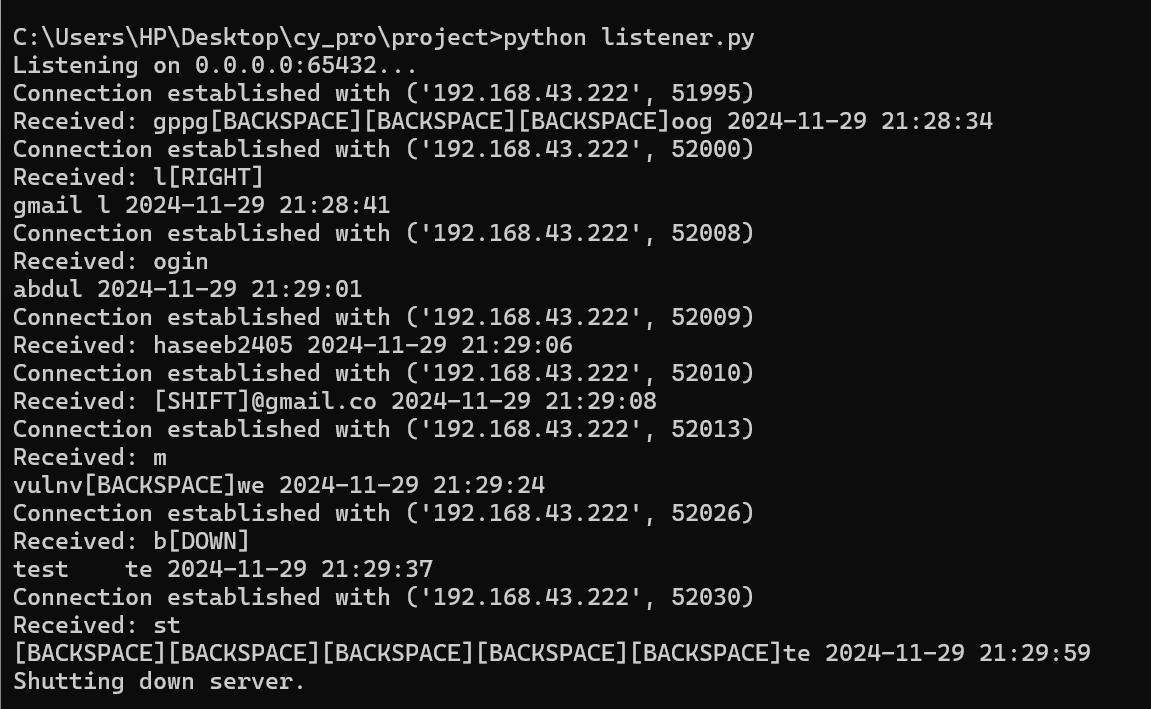
When executed, this script generates infinite popup windows, rendering the system **unusable** and **forcing** the victim to **step away**.

### ****Step 7: Listening for Logs****

On the attacker’s machine, a socket server is set up to listen for incoming keystrokes:

//saved as listener.py

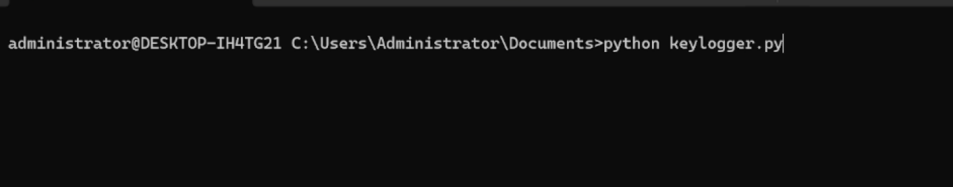
This server continuously receives after the execution of keylogger.py script and logs the keystrokes transmitted from the victim's machine.

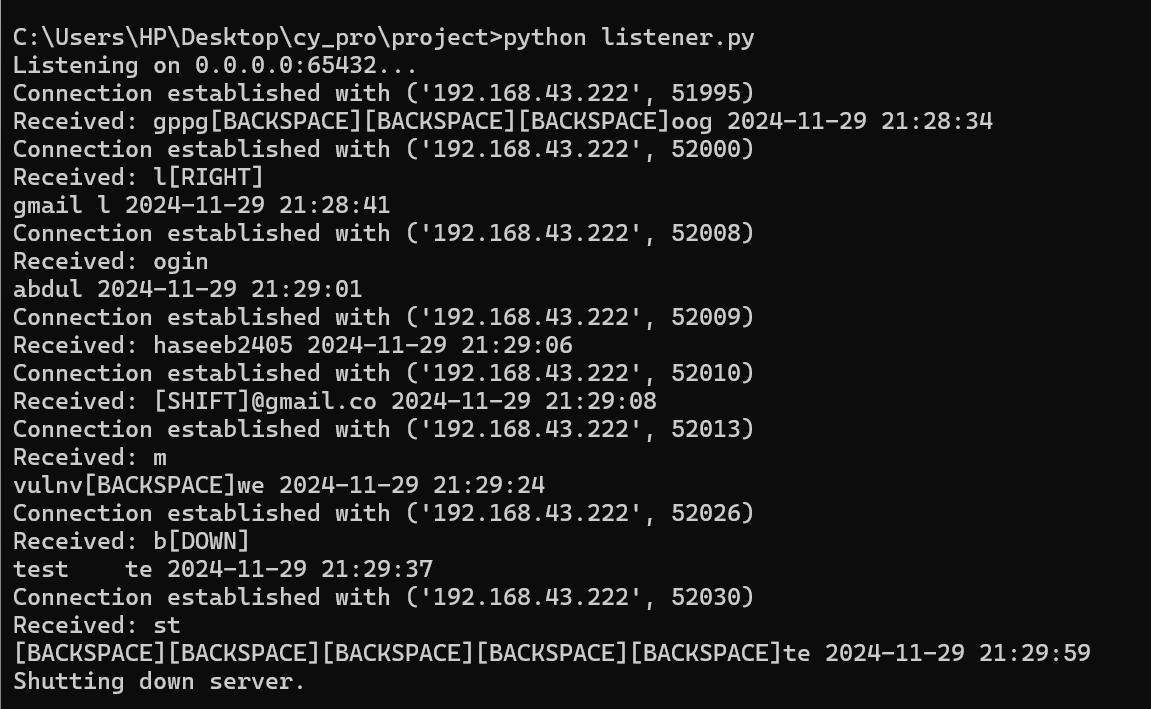
### ****Step 7: Execution of Script and Receiving of Keystrokes.****

****On console:****

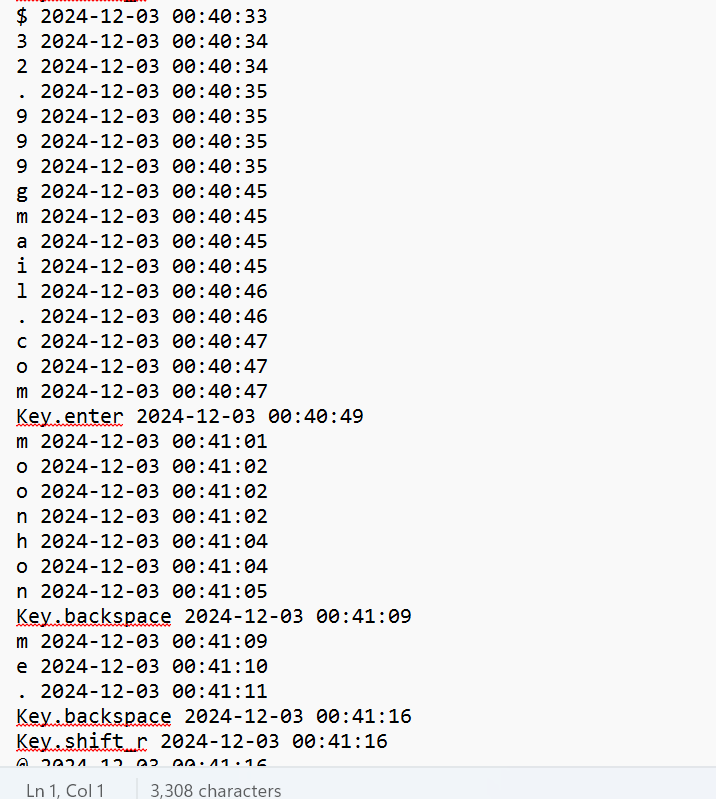
****Target Pc shell****



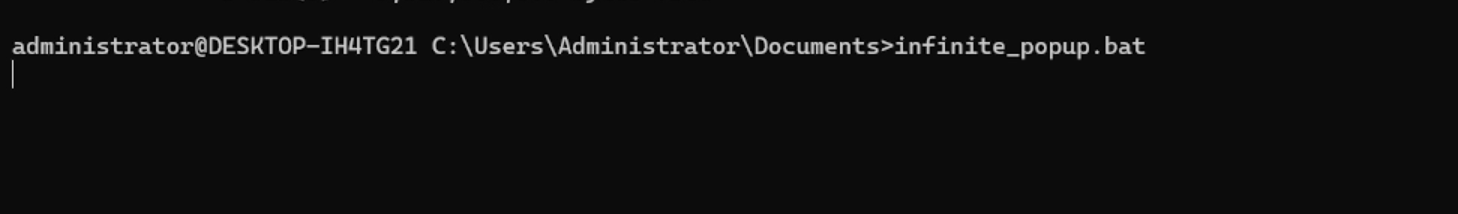
**Attacker listening and recording logs**

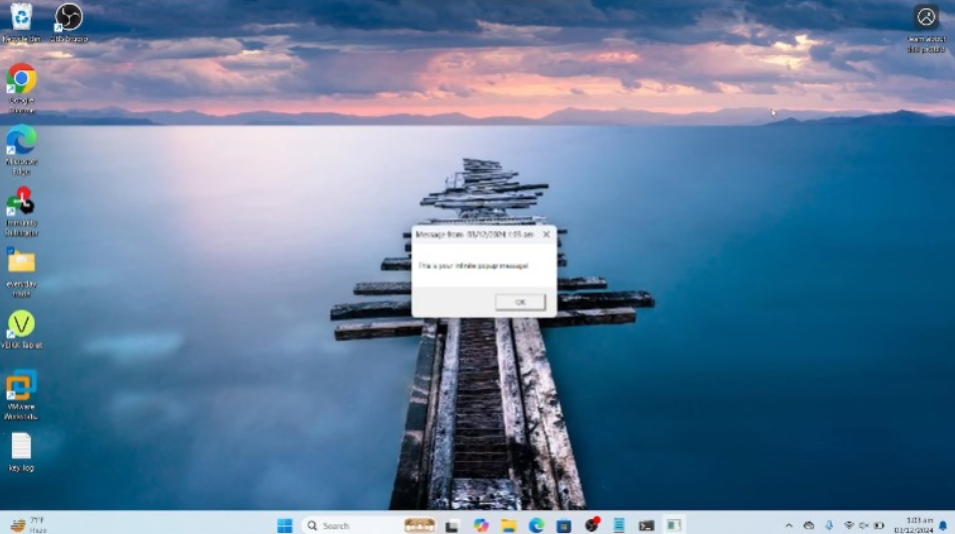


**Saved in logs.txt file:**



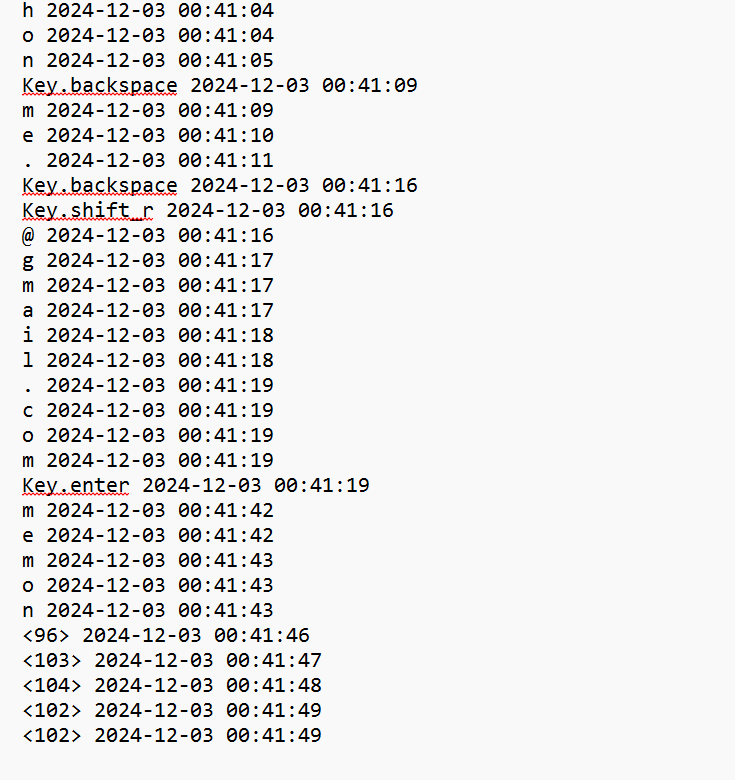
### ****Step 8: Executing error.bat file to make target distracted.****





### ****Step 9: Credential Analysis****

While the victim is distracted, the attacker analyzes the logs.txt file to extract sensitive information, such as email credentials. If the data is encoded(like here numeric keys were representing other numbers), decoding techniques are applied to retrieve plaintext information.

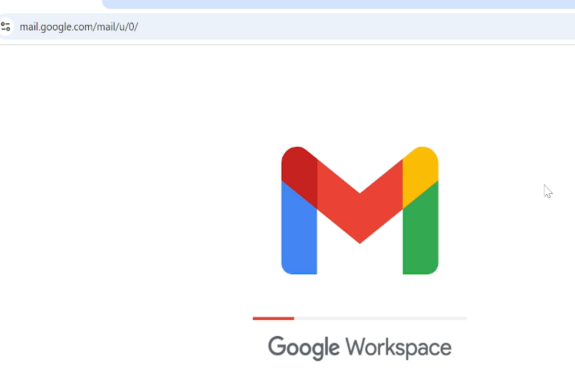
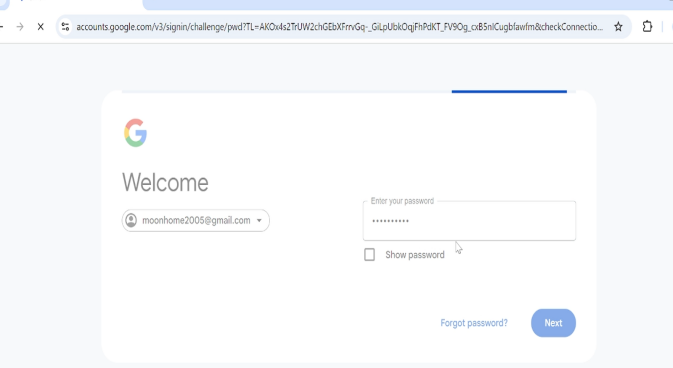


//assuming attacker has mapping strategies.



### ****Step 10: Unauthorized Account Access****

Using the harvested credentials, the attacker logs into the victim’s Gmail account to perform unauthorized actions.



## ****Project Outcomes****

The project successfully demonstrated:

1. **Network Scanning**: Identifying vulnerable devices on a local network.
2. **Brute-Force Techniques**: Exploiting weak SSH credentials.
3. **Payload Deployment**: Delivering and executing a Python-based keylogger.
4. **Distraction Tactics**: Using malicious scripts to disrupt the victim’s activities.
5. **Credential Harvesting**: Analyzing keystroke logs to extract sensitive information.

## ****Limitations:****

* Attack was performed in a controlled environment.
* This was only possible on local network,as on public network there are various factors considering firewalls,keystrokes logging over the public network that needs greater hardware

tools that was not possible for us.

* Script execution might be restricted on various OS due to securtiy policies implemented as during testing we encountered this issue.
* SSH connection even on local network can cause issues on being connected due to port 22 not being open by default.

## ****Mitigation Strategies****

**Use Strong Passwords**

* Enforce password policies requiring complex, unique passwords.
* Periodically update passwords to reduce the risk of brute-force attacks.

**Secure Remote Access**

* Disable unnecessary services such as SSH on non-administrative systems.
* Use key-based SSH authentication instead of password-based access.

**Implement Network Monitoring**

* Deploy intrusion detection systems (IDS) to identify suspicious activities.
* Regularly scan the network for unauthorized devices.

**Enable Two-Factor Authentication (2FA)**

Add an extra layer of security to critical accounts.

**Educate Employees**

* Train staff to recognize suspicious activity and practice safe computing habits.

**Endpoint Security Solutions**

* Use endpoint detection and response (EDR) tools to monitor and block malicious scripts.
* Keep antivirus software updated to detect and mitigate threats.

## ****Conclusion****

This project highlights the working of a keylogger script,vulnerabilities in network security and emphasizes the importance of proactive measures to safeguard against attacks. By simulating this attack in a controlled environment, we underline the critical need for organizations to implement robust security practices and educate employees to mitigate risks effectively.