**CMPE494 FREE PROJECT REPORT**

**Students: Kursat Bakis, Muhsin Etki**

This project is a virus written in Python. It aims to take advantage of Windows systems’ security vulnerabilities and python’s strong ability to interact with operating system. It comprises of multiple components:

**-Wi-Fi Passwords Stealer:**

The program fetches the names and passwords of all the wi-fi networks the computer has connected until now. It writes the results to the file ‘wifiPasswords.txt’.

**-Chrome Saved Passwords Stealer:**

The program connects to Google Chrome’s database that contains all the saved passwords via sqlite3. The database path in Windows computers is:

C:\Users\Msi\AppData\Local\Google\Chrome\User Data\Default\Login Data

But the passwords in Chrome are kept encrypted by AES algorithm. So, we decrypt using Cipher object of Python’s Cryptography library. There is also the possibility of it being encrypted by a different algorithm, in case it was used relatively older. If we detect such situation, ctypes library and winDLL api of python is used. All the websites, usernames and passwords are written into ‘chromePasswords.txt’.

**-Webcam:**

Takes 3 photos of you every 2 seconds by turning on your webcam. Saves the photos into the files ‘camera1.jpg’ and so on. It uses the cv2 library of python. It also captures two screenshots of your screen, using pyautogui library.

**-Keylogger:**

Name is explanatory enough, it logs the keyboard strokes of the user into ‘log.txt’. It is possible to do this by pynput.keyboard library.

**-Main:**

It starts the keylogger as a different thread, then calls all the functions that are explained above. After the fetching is done for all of them, and the files are ready, it sends the files to my server on Microsoft Azure by a POST request to IP: (<http://40.115.36.79/upload>)

I connected my own computer to remote via SSH, and can access all the files in it.

**-Server:**

We wrote a simple Flask server in server.py file. It runs on remote computer, in order to receive POST requests from victims. When a POST request that contains the files arrives in the server, it saves the files into upload folder that we assigned.

**How to run:**

First, install the packages that are imported (You can use requirements.txt file).

Then, run main.py from terminal.

\*\*\*BE CAREFUL!! WHEN YOU RUN MAIN.PY, YOUR PASSWORDS AND SCREENSHOTS MIGHT BE SENT TO OUR SERVER!! DO IT AT YOUR OWN RISK!!

\*\*You don’t have to run the server.py file. Because it runs continuously on the server’s IP.

\*\*We took help from GitHub’s open repositories which contain similar functionalities with ours.

\*\*Right now, our virus requires the victim’s computer to have python installed. But you can easily turn all these files into a single executable so it will work when you double-click.