ASSIGNMENT 4 AS FINAL

CENG3544, COMPUTER AND NETWORK SECURITY

Hasibullah Mahmood
hasibullahmahmood@posta.mu.edu.tr

Monday 8th June, 2020

1 Write your thoughts about Malware (Antivirus etc.) tools (write in max 100 words) (10 pts.)

Here I would like to write briefly about anti-virus, anti-malware and their advantages and disadvantages.
The Anti-virus software is used to detect, quarantine and remove viruses from the system. Anti-virus usually deals with older and identified threats like Trojans, viruses and worms. On the other hand, anti-malware usually deals with new threats or delivered by zero-day exploits.

1.1 Advantages:

1. Virus protection
2. Spyware protection
3. Web protection
4. Spam protection
5. Firewall feature

1.2 Disadvantages

1. System slowdown
2. No complete protection
3. Untrustworthy

Today both premium and free anti-viruses are available in markets. For premium antiviruses, you need to pay monthly or yearly. However, for free antiviruses as the name suggests there is no fee but advertisement.

2 Why is Linux preferred in SOC as SIEM? (Write in max 100 words) (10 pts.)

Linux operating system is very flexible and we can configure it based on our needs. We can install just the required programs and uninstall extra unnecessary programs, so the operating system will run fast and smooth.
Another feature of Linux OS is the existence of ready tools for Security information and event management (SIEM). Some of them are listed as follows:

1. Network packet capture software like Wire-shark
2. Malware analysis tools
3. Intrusion detection systems (IDSs)
4. Firewalls
5. Log managers
6. And ticketing systems

These great features made Linux the best candidate as an operating system for the security operation center (SOC).

3 Write and enhance your Linux script in the Midterm Assignment so that it (60 pts.)

1. (a) Gives a summary report of the system to file(s)
   (b) Periodically runs
   (c) Keeps the current status of the system in USB (or cloud?) (BONUS)
   (d) Uses hashing
   (e) Encrypts the current status (file(s))with your key)

A script is written and attached.
Comments are provided for almost all the lines of codes.

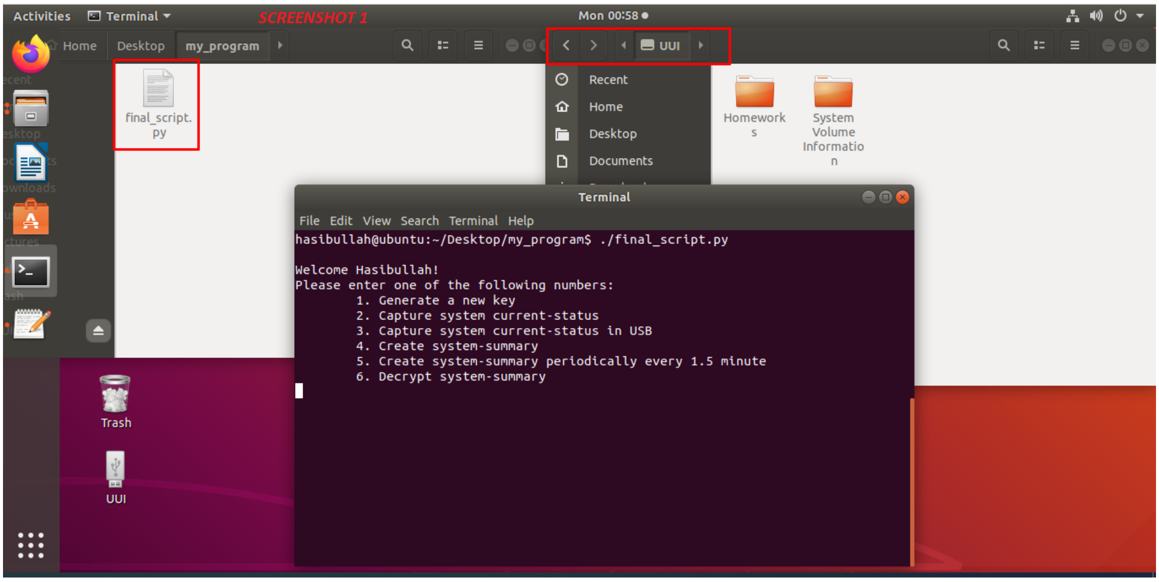
4 Document your script and your findings in your report (20 pts.)

Screenshots and detailed explanation of the written code should be included.

Screenshots and detailed explanation:

As shown below in SCREENSHOT 1, the script is run. It first gets the username and prints some options to choose from. The user can enter one of 6 numbers based on his/her need and the script will continue to execute.

Also, looking to SCREENSHOT 1, we can see there are two file explorer opened. On the left-hand side, the script is located in the local disk. On the right-hand side, there is a USB named UUI.



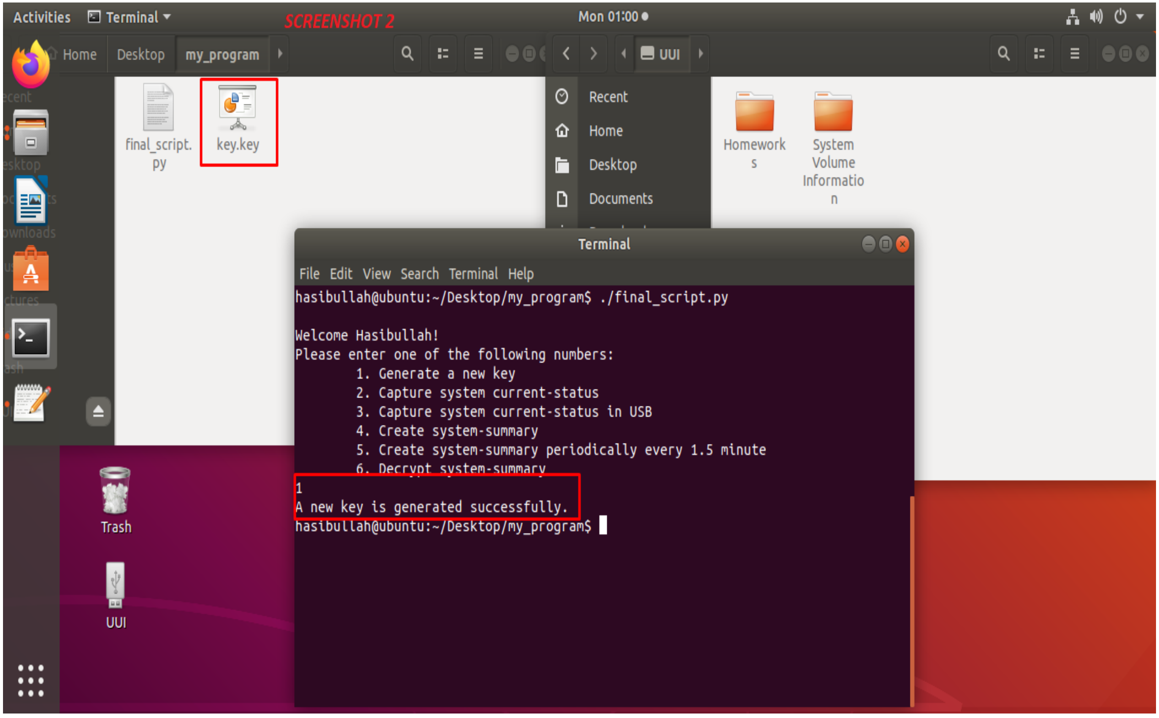
```
hasibullah@ubuntu:~/Desktop/my_program$ ./final_script.py

Welcome Hasibullah!
Please enter one of the following numbers:
        1. Generate a new key
        2. Capture system current-status
        3. Capture system current-status in USB
        4. Create system-summary
        5. Create system-summary periodically every 1.5 minute
        6. Decrypt system-summary
```

In the beginning, as a user, I didn't have a key so I entered 1 to create it. Then the key is generated in the local disk.

Note: The key is not generated, unless the user enters 1. If the user already has a key, he/she can skip the first option and continue with the rest of the options.

Also, after the key is created, the user can store the key in any safe place.



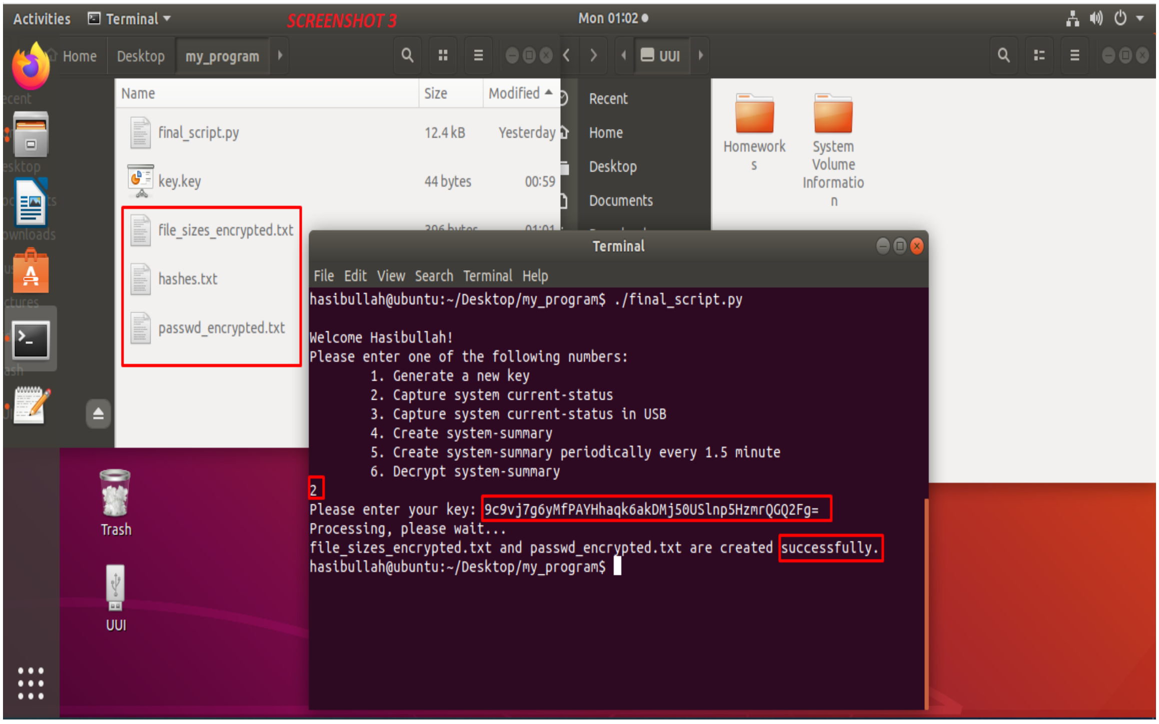
```
hasibullah@ubuntu:~/Desktop/my_program$ ./final_script.py

Welcome Hasibullah!
Please enter one of the following numbers:
        1. Generate a new key
        2. Capture system current-status
        3. Capture system current-status in USB
        4. Create system-summary
        5. Create system-summary periodically every 1.5 minute
        6. Decrypt system-summary
1
A new key is generated successfully.
hasibullah@ubuntu:~/Desktop/my_program$
```

In the 3^{rd} screenshot, option 2 is selected. The script asks for the key to be entered manually. The user is free to type it or just copy past it. As we can see, three new files are generated in the local disk, two of them encrypted and the third file is their hash after encryption.

Note that, first I have encrypted the files and then compute their hashes because the hash is one way and the original file can't be generated by it.

Also, the user is free to keep the hash file in any safe place.

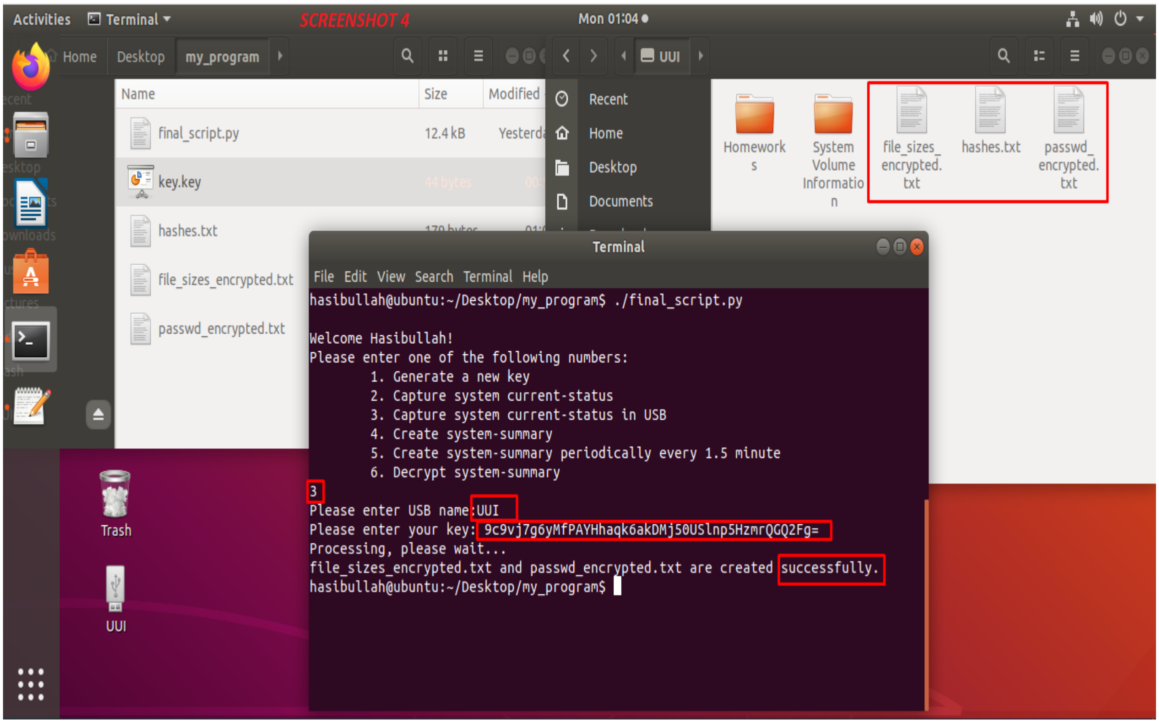


```
hasibullah@ubuntu:~/Desktop/my_program$ ./final_script.py

Welcome Hasibullah!
Please enter one of the following numbers:
        1. Generate a new key
        2. Capture system current-status
        3. Capture system current-status in USB
        4. Create system-summary
        5. Create system-summary periodically every 1.5 minute
        6. Decrypt system-summary
2
Please enter your key: 9c9vj7g6yMfPAYHhaqk6akDMj50USlnp5HzmrQGQ2Fg=
Processing, please wait...
file_sizes_encrypted.txt and passwd_encrypted.txt are created successfully.
hasibullah@ubuntu:~/Desktop/my_program$
```

The 4^{th} screenshot shows that the user has selected option 3. Then, the user is asked to enter the USB name and key.

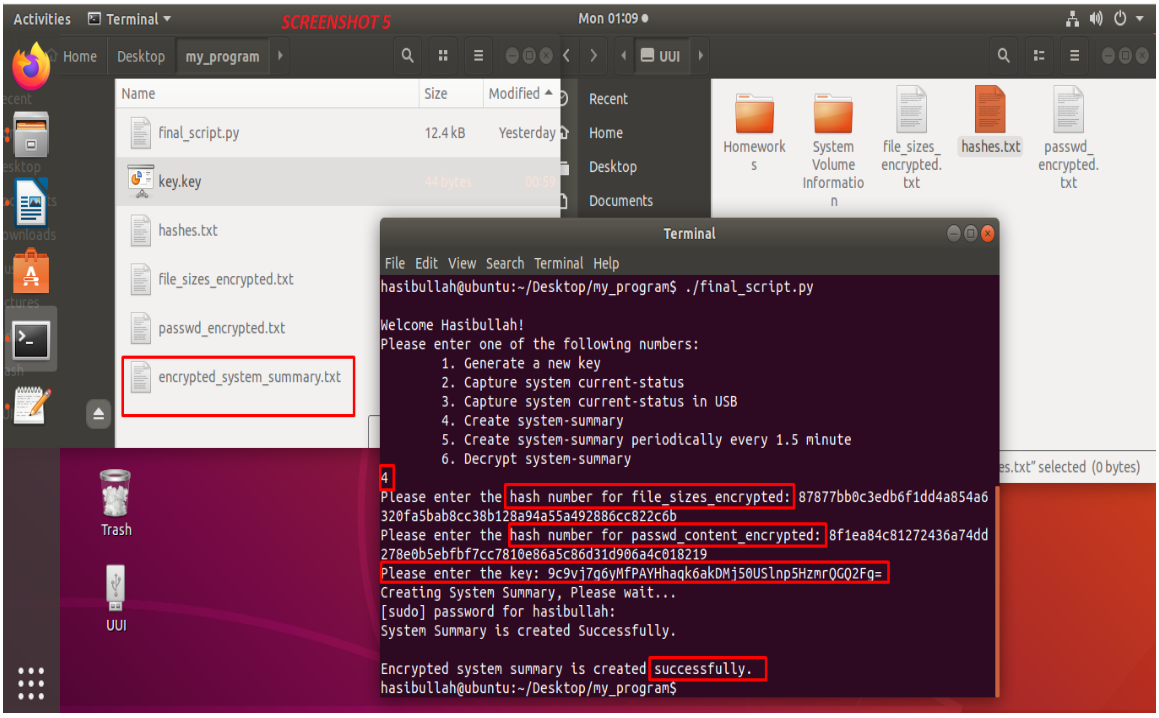
The script will check whether the given USB exists or not. If the USB doesn't exist, it will print a message as "USB not found". Else the key will be asked and the system status files will be generated.



```
hasibullah@ubuntu:~/Desktop/my_program$ ./final_script.py

Welcome Hasibullah!
Please enter one of the following numbers:
        1. Generate a new key
        2. Capture system current-status
        3. Capture system current-status in USB
        4. Create system-summary
        5. Create system-summary periodically every 1.5 minute
        6. Decrypt system-summary
3
Please enter USB name:UUI
Please enter your key: 9c9vj7g6yMfPAYHhaqk6akDMj50USlnp5HzmrQGQ2Fg=
Processing, please wait...
file_sizes_encrypted.txt and passwd_encrypted.txt are created successfully.
hasibullah@ubuntu:~/Desktop/my_program$
```

If the user enters 4 to create a system summary, the user will be asked to manually enter or copy past hashes. Here, the script will check whether system status files exist or not. If the files are not found, a message like “The system files not found” error will be printed. Else, the user again will be asked to enter the key and finally, the system summary will be generated.

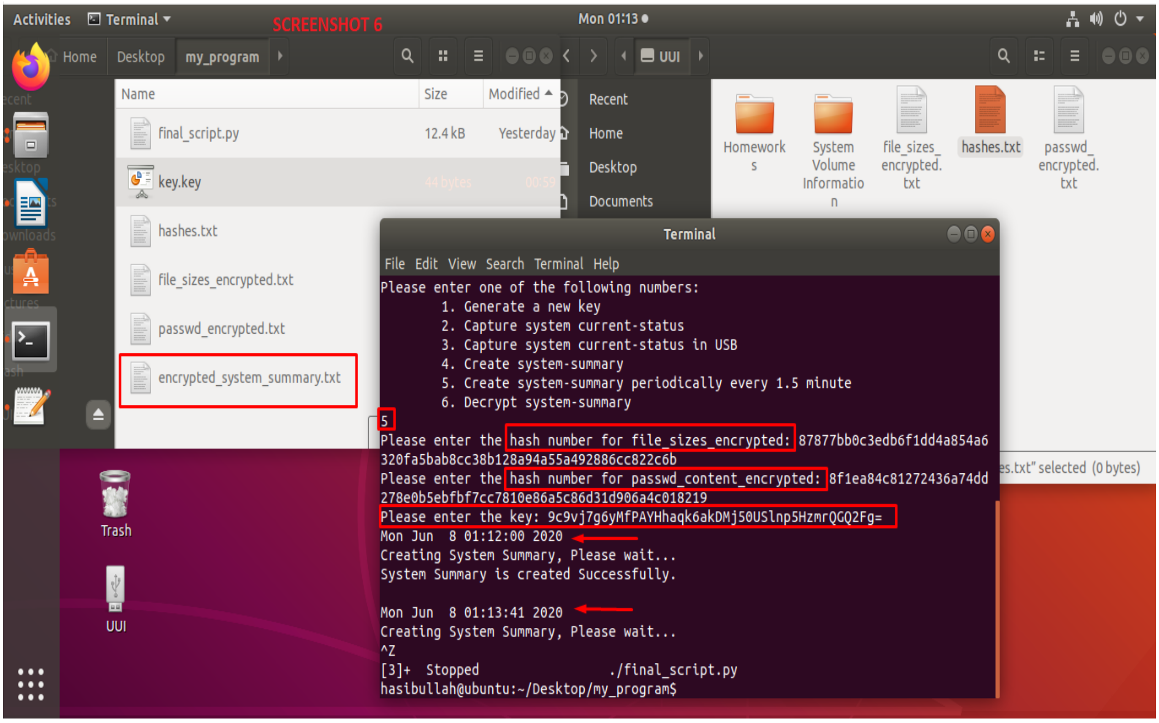


```
hasibullah@ubuntu:~/Desktop/my_program$ ./final_script.py

Welcome Hasibullah!
Please enter one of the following numbers:
        1. Generate a new key
        2. Capture system current-status
        3. Capture system current-status in USB
        4. Create system-summary
        5. Create system-summary periodically every 1.5 minute
        6. Decrypt system-summary
4
Please enter the hash number for file_sizes_encrypted: 87877bb0c3edb6f1dd4a854a6
320fa5bab8cc38b128a94a55a492886cc822c6b
Please enter the hash number for passwd_content_encrypted: 8f1ea84c81272436a74dd
278e0b5ebfbf7cc7810e86a5c86d31d906a4c018219
Please enter the key: 9c9vj7g6yMfPAYHhaqk6akDMj50USlnp5HzmrQGQ2Fg=
Creating System Summary, Please wait...
[sudo] password for hasibullah:
System Summary is created Successfully.

Encrypted system summary is created successfully.
hasibullah@ubuntu:~/Desktop/my_program$
```

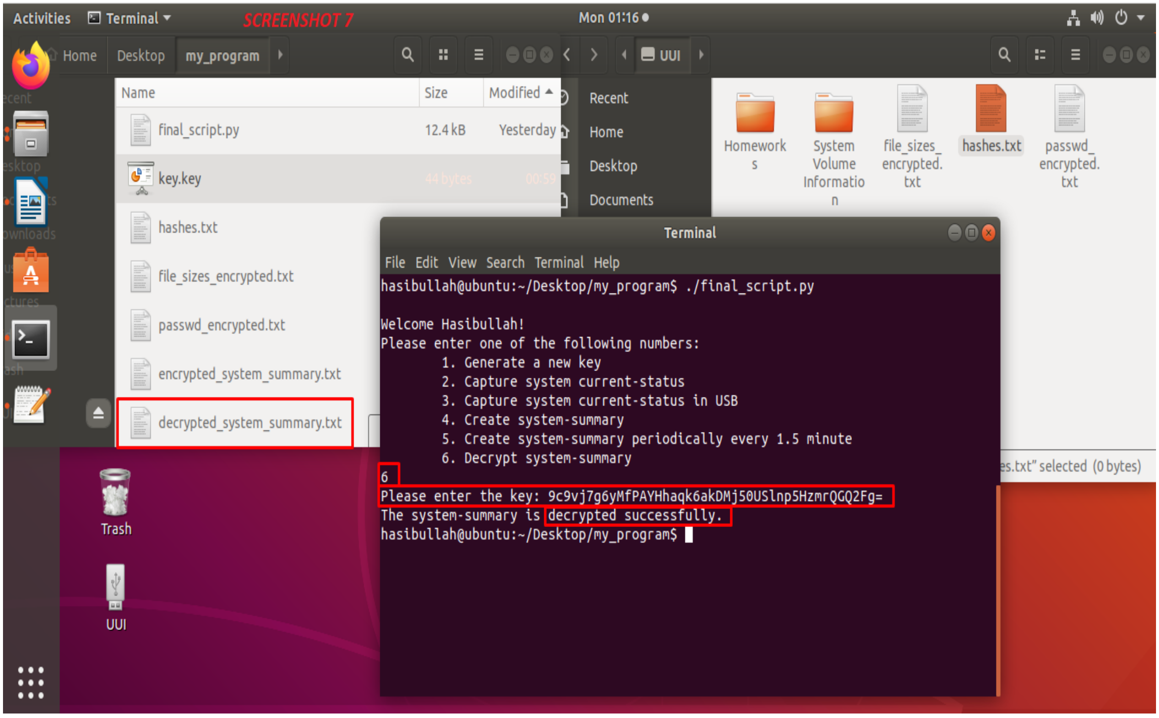
Furthermore, if the user enters 5, the user will be asked to enter hashes and key and as a result, the system summary will be prepared, encrypted and appended to the file.



```
Please enter one of the following numbers:
        1. Generate a new key
        2. Capture system current-status
        3. Capture system current-status in USB
        4. Create system-summary
        5. Create system-summary periodically every 1.5 minute
        6. Decrypt system-summary
5
Please enter the hash number for file_sizes_encrypted: 87877bb0c3edb6f1dd4a854a6
320fa5bab8cc38b128a94a55a492886cc822c6b
Please enter the hash number for passwd_content_encrypted: 8f1ea84c81272436a74dd
278e0b5ebfbf7cc7810e86a5c86d31d906a4c018219
Please enter the key: 9c9vj7g6yMfPAYHhaqk6akDMj50USlnp5HzmrQGQ2Fg=
Mon Jun  8 01:12:00 2020
Creating System Summary, Please wait...
System Summary is created Successfully.

Mon Jun  8 01:13:41 2020
Creating System Summary, Please wait...
^Z
[3]+  Stopped                 ./final_script.py
hasibullah@ubuntu:~/Desktop/my_program$
```

Finally, if the user enter 6, the script will prompt for key and as a result, the decrypted system summary as a plaintext will be generated.



```
hasibullah@ubuntu:~/Desktop/my_program$ ./final_script.py

Welcome Hasibullah!
Please enter one of the following numbers:
        1. Generate a new key
        2. Capture system current-status
        3. Capture system current-status in USB
        4. Create system-summary
        5. Create system-summary periodically every 1.5 minute
        6. Decrypt system-summary
6
Please enter the key: 9c9vj7g6yMfPAYHhaqk6akDMj50USlnp5HzmrQGQ2Fg=
The system-summary is decrypted successfully.
hasibullah@ubuntu:~/Desktop/my_program$ 
```