# ENPM687 – Digital Forensics and Incidence Responses

# Final Report

*Author – Syed Mohammad Ibrahim*

*UMD ID:* *iamibi*

*UID: 118428369*

*Email:* [*iamibi@umd.edu*](mailto:iamibi@umd.edu)

## Brief Summary of Information

The goal of this document is to go over the Rebel malware hard drive recovered by the Imperial Army. This report will outline the findings and the steps involved in achieving them. The malware has reportedly been connecting with its command-and-control center to transmit information that might cause problems to the Imperial Army. The investigation performed as part of forensics found the plans for the Death Star which can help the Rebel army to initiate a counterattack on Imperial Army.

## Tools Used in the investigation

* Autopsy
* WireShark
* VeraCrypt

## Repository

The following steps were involved in the forensics investigation which yielded the plans for the Death Star that were sent by the malware to C2:

1. The hard drive image provided was of windows XP. The image was loaded in the Autopsy, and it generated the file system hierarchy.
2. While traversing the directories in the “vol2”, I found that under “Documents and Settings/Administrator/Recent” there were a few interesting entries like “obiwan.lnk”, “obiwan2.lnk”. Also, there were a few files that said “plans” in their names like “the-official-rogue-one-death-star-plans-are-revealed.lnk”, etc.
3. Then I proceeded to traverse other folders and found the binaries named “obiwan.exe” and “obiwan2.exe” at the location “Documents and Settings/Administrator/My Documents/code/dist/”. These binaries were generated by the python package called pyinstaller as there were “spec” files present.
4. Along with the binaries, in a folder up, there was a source file called “obiwan.py” which contained the source code of the assumed obiwan.exe binary. However, there were two different binaries found, which questioned the piece of code legitimacy. The only way to confirm is to run them in an isolated environment and track their network calls.

Graphical user interface, text, application, email

Description automatically generated

1. The binary “obiwan2.exe” was run in an isolated environment with WireShark monitoring the network traffic. It made the calls shown in the below picture:

Graphical user interface, application

Description automatically generated

1. The GET request “/All-your-base64-are-belong-to-us” pointed out that there is a presence of base64 type string. Looking at another GET request “/cjJkMiBpcyB0aGUga2V5”, it seemed that this is a base64 string. So, I decoded it with python’s base64 decoder.

Graphical user interface, text, application, email

Description automatically generated

1. This indicated that there is potential an encrypted text or file which will require this key to be used. I searched the image in autopsy again and found few files which autopsy indicated as encrypted. However, I didn’t know what kind of tool or encryption algorithm was used to encrypt them. I searched other folders and found a VeraCrypt tool binary. After googling, I found that it is an encryption utility tool which I believed was used to encrypt these files.

Graphical user interface, text, application, email

Description automatically generated

1. I downloaded the tool, installed, and ran it on the files that I extracted with the “r2d2” key. Only one file was successfully decrypted named “3066-not-the-droids-youre-looking-for.mp3”. I mounted it on virtual drive “Z”.

A screenshot of a computer

Description automatically generated with medium confidence

1. I read the ENPM687-Read-This.txt file which stated that the final version of the malware “final-form.exe” has sent few messages. I ran the binary in an isolated environment with WireShark tracking the network calls and found the following

Graphical user interface, text, application

Description automatically generated

1. The final message sent by the Rebel malware was captured and there were plans present in the mounted drive with title “Death Star Plans”.

Graphical user interface

Description automatically generated

## Recommendations

The final message recovered has made the Rebel army aware of the Death Star plans. It is recommended that the Imperial Army add security to their communication channels and make sure that such detail plans are encrypted with a better key length (minimum 15 characters). The key should also be stored in a secure fashion as Base64 is a weak scrambling algorithm. One can always use a Password Manager for storing such sensitive passwords.

## Challenges during investigation

1. It took a lot of time to figure out what kind of tool or utility was used to encrypt the files. Googling a lot of things that can potentially be a tool or using a python script to encrypt was a big blocker for me. Eventually, I landed on veracrypt binary and its details.
2. The veracrypt is not an easy-to-use tool. I had to search and read documentations on how to use it to finally figure it out.