

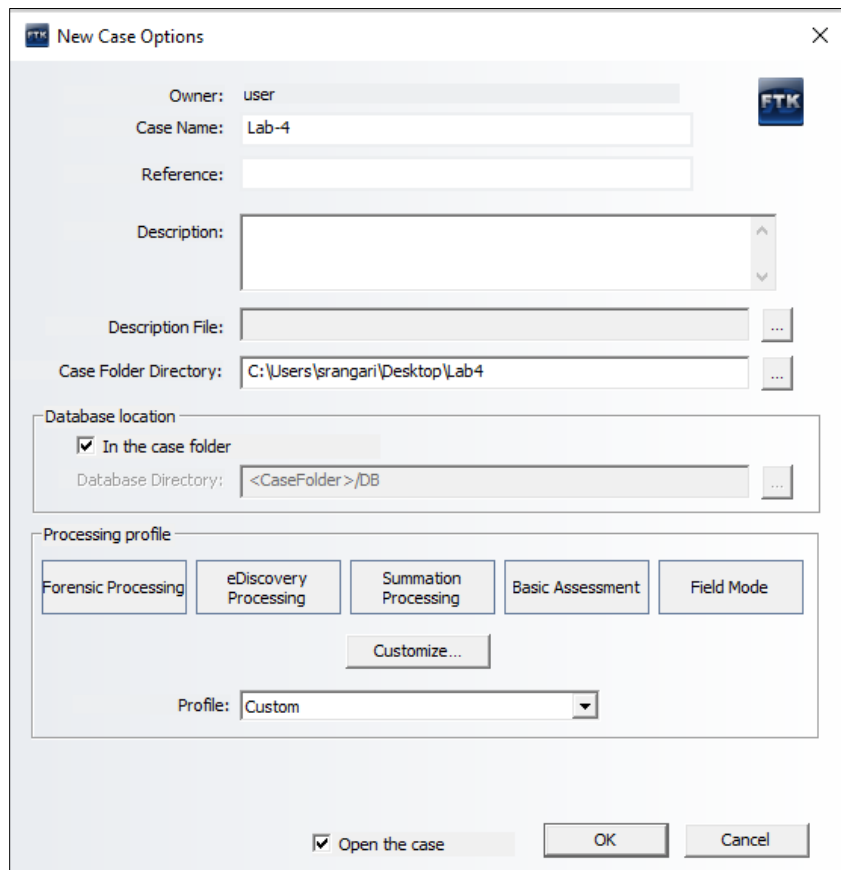
ITIS 5250
Sneha Rangari
Lab 4
10/29/2018

Overview:

In this Lab, I have been given one file, namely “MyFiles.iso”. I have been asked to make use of the “FTK Tool” along with “SilentEye Tool” to find any information to show that someone is stealing our technology. Thus I will be finding certain hidden information from the given image. Also, I will be extracting hash value of the evidence.

Forensic Acquisition & Exam Preparation:

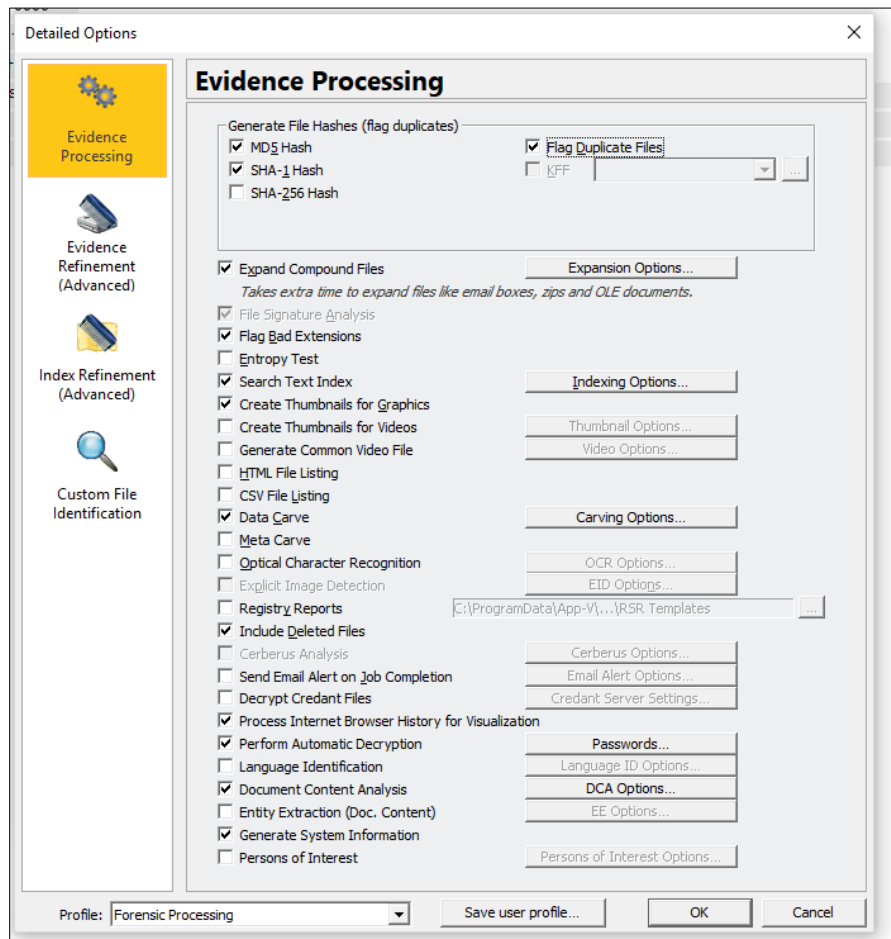
I accessed the Forensic images in the Shared Folder on the network from the Forensics Lab in Cone 169. I accessed iso file named “MyFiles.iso”. The software used for accessing & extracting information from the image is FTK Tool. The first step undertaken after accessing the file was loading the image using FTK for evidence processing.



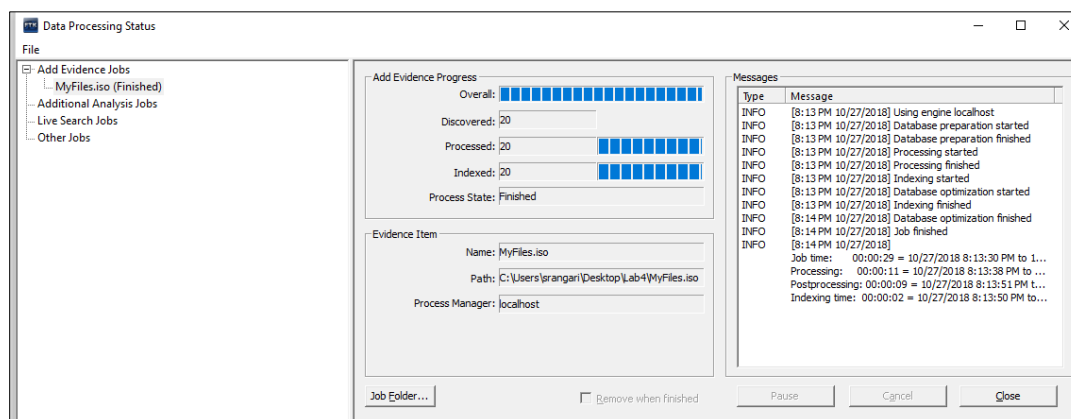
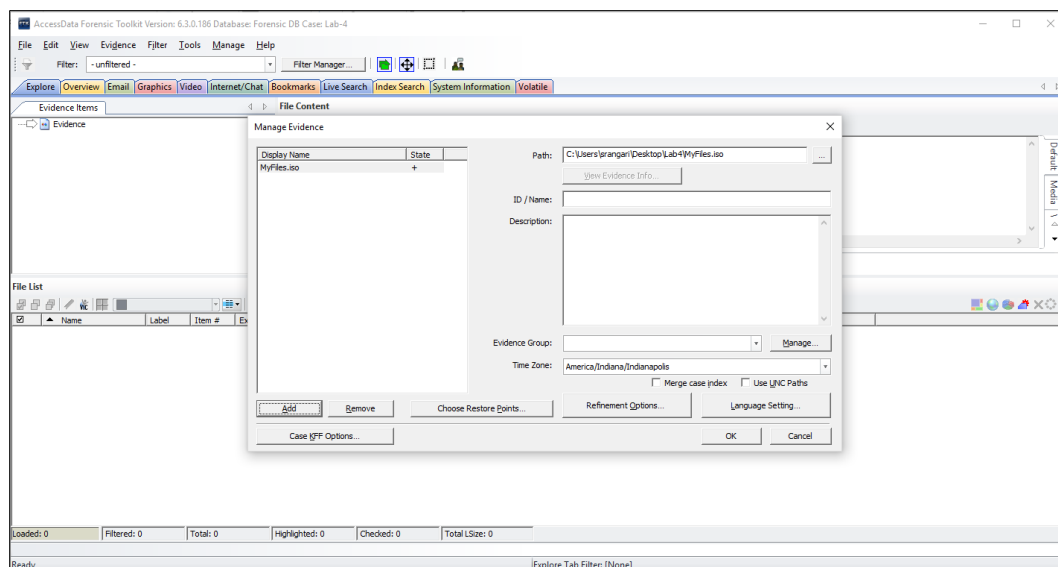
Where I customized processing profile by adding options like:

- Data Carve
- Include Deleted Files

- Process Internet Browser History for Visualization
- Perform Automatic Decryption
- Document Content Analysis
- Generate System Information.

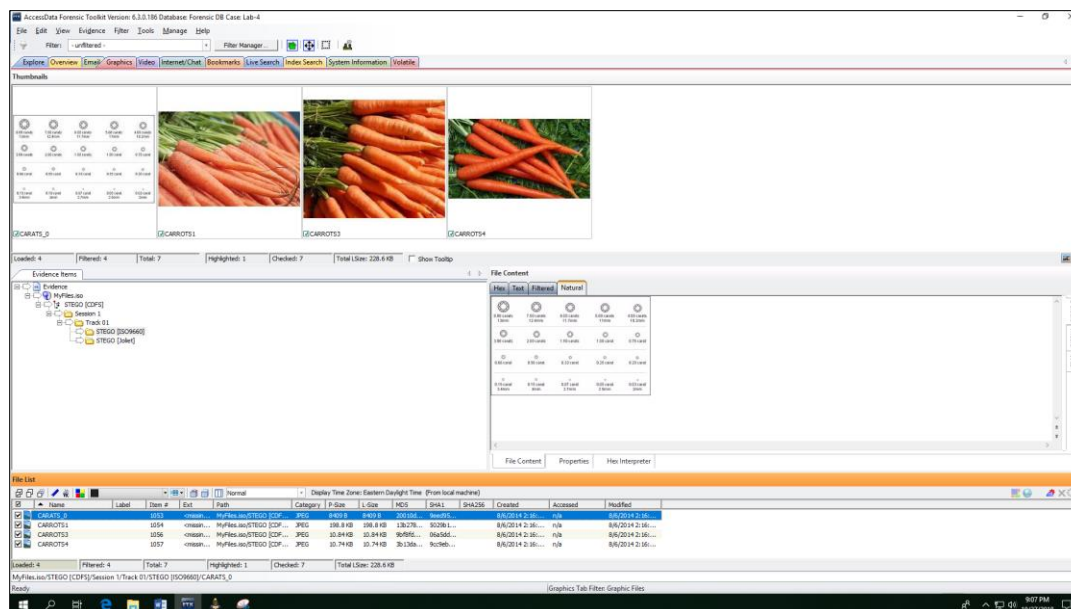


Thus I created a new case in FTK by loading a MyFiles.iso for evidence processing.



Findings & Report (Forensic Analysis)

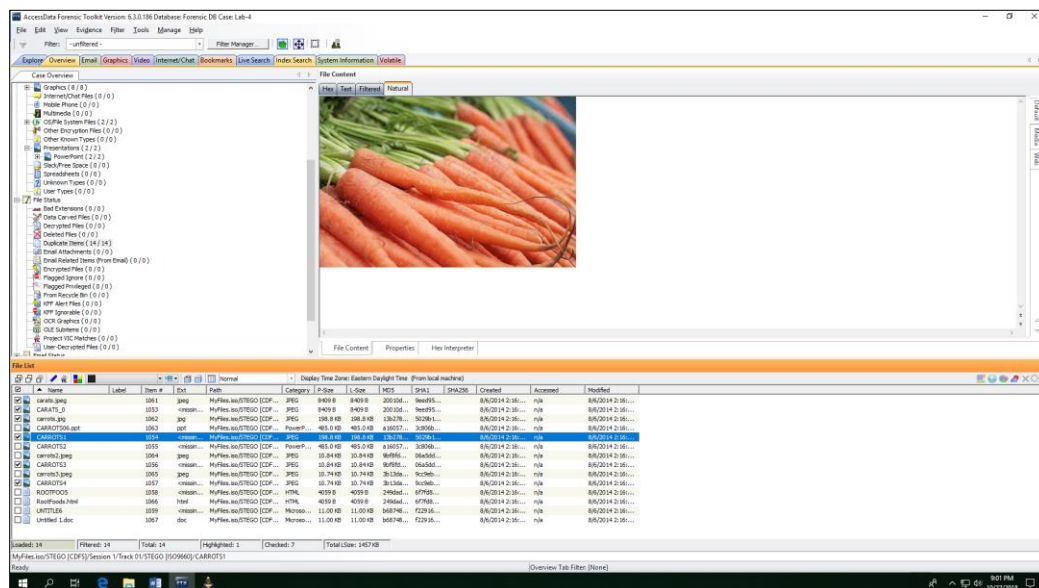
After loading the image into FTK, I found various images of Carrots in graphics tab for loaded image.



Can you confirm any hidden information relating to the recovery of the schematic?

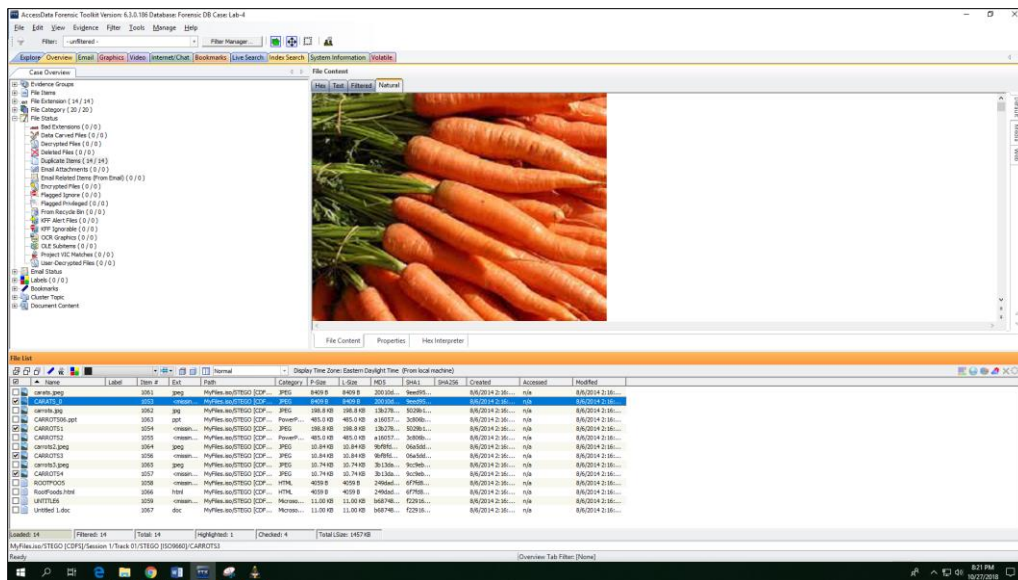
After going into Overview tab and searching in folders, I found following documents.

CARROTS 1:

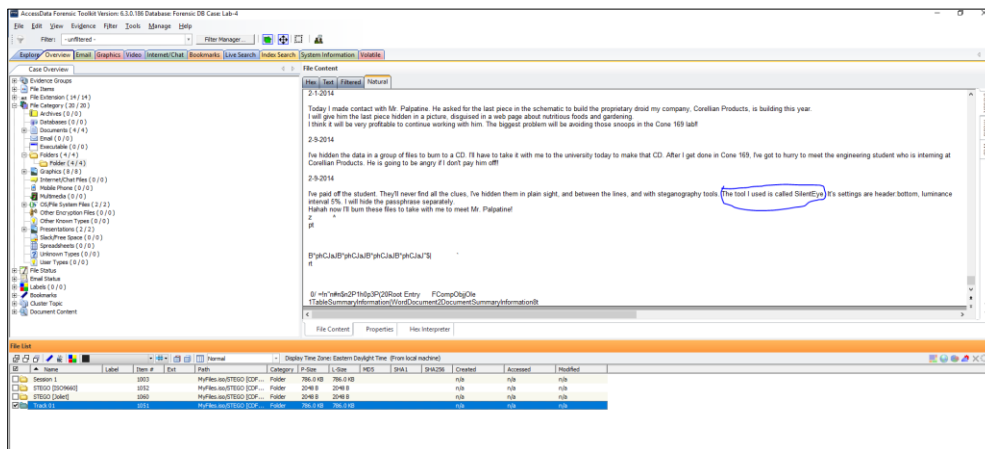


CARROTS 3:

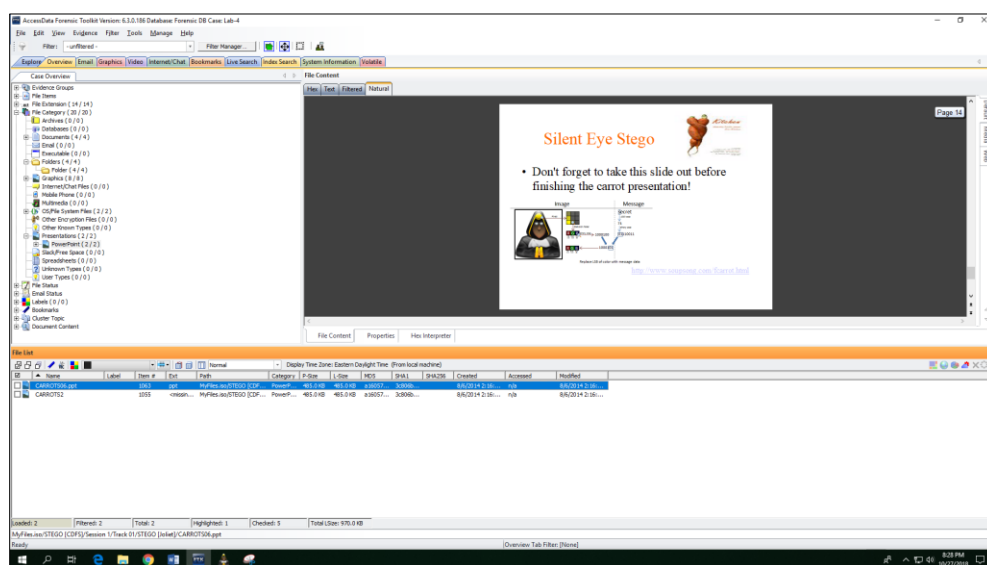
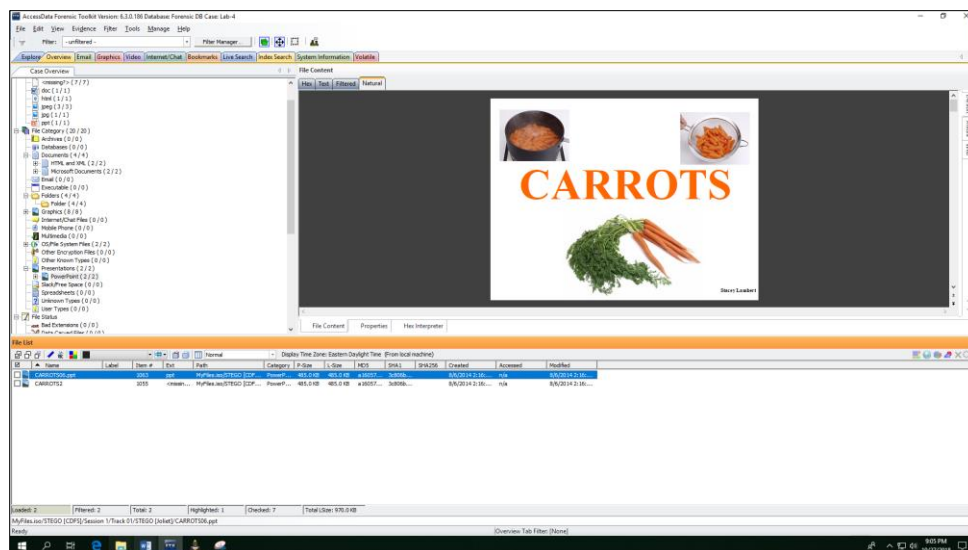




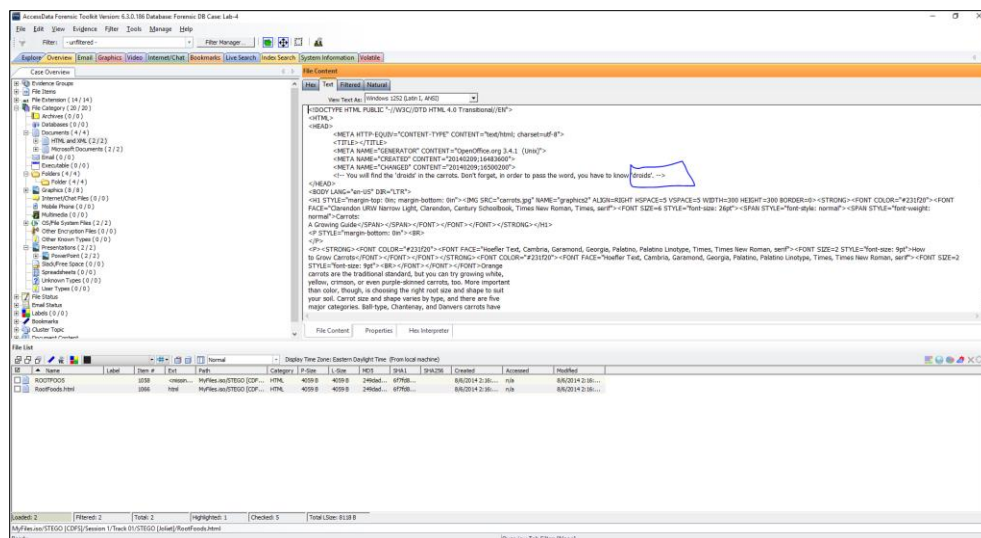
After investigating in folders, I found that the tool used for steganography is SilentEye.



Also in PowerPoint folder, I got confirmation on use of SilentEye Tool which is a cross-platform application design for an easy use of steganography, in this case hiding messages into pictures or sounds. It provides a interface and an easy integration of new steganography algorithm and cryptography process by using a plug-ins system.

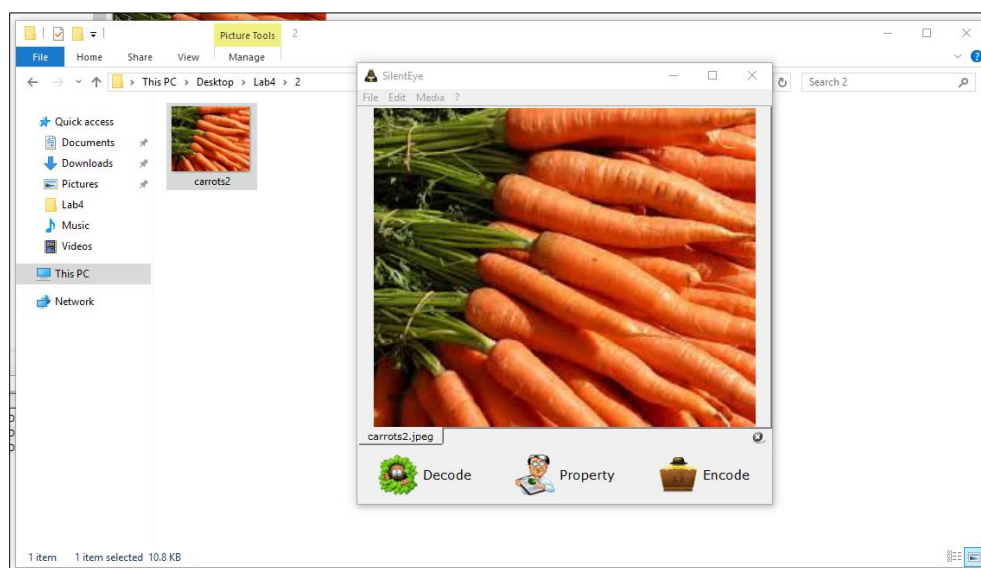


On further investigation in documents, I found that the passphrase for SilentEye Tool to decode the image is “**droids**” as shown below:

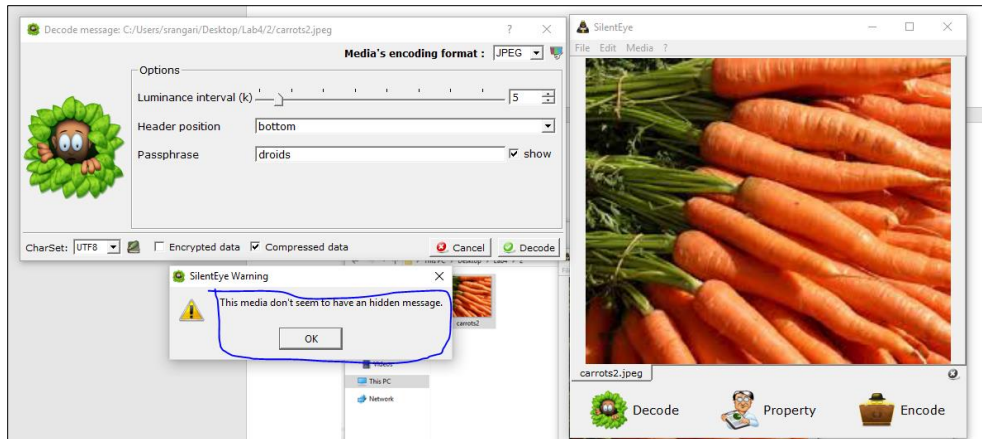


SilentEye Tool:

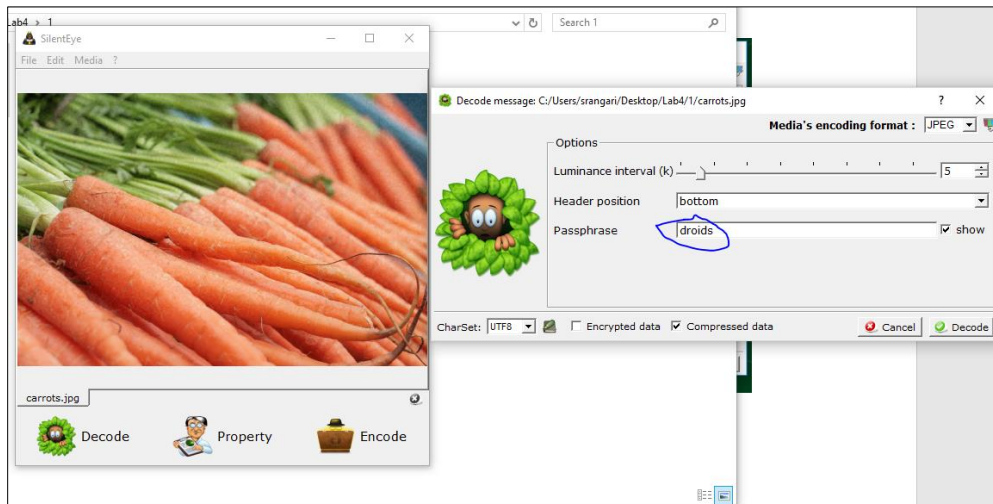
I opened SilentEye Tool and loaded the **carrots2** jpeg as shown:



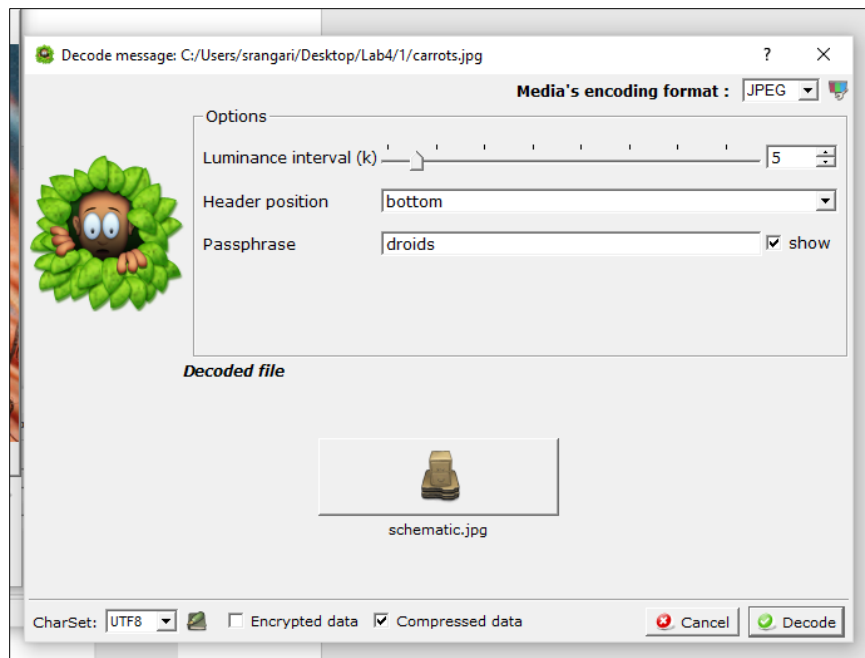
After entering passphrase as droids, I started to decode it but there was **no hidden message** in that jpeg file as shown:



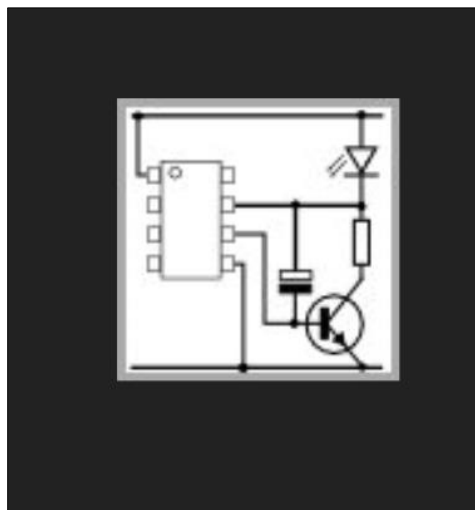
Later, I opened SilentEye Tool and loaded the **carrots** jpeg as shown:



Again after entering passphrase as droids, I started to decode it and there **was hidden message of schematic.jpg** in that carrots.jpeg file as shown:



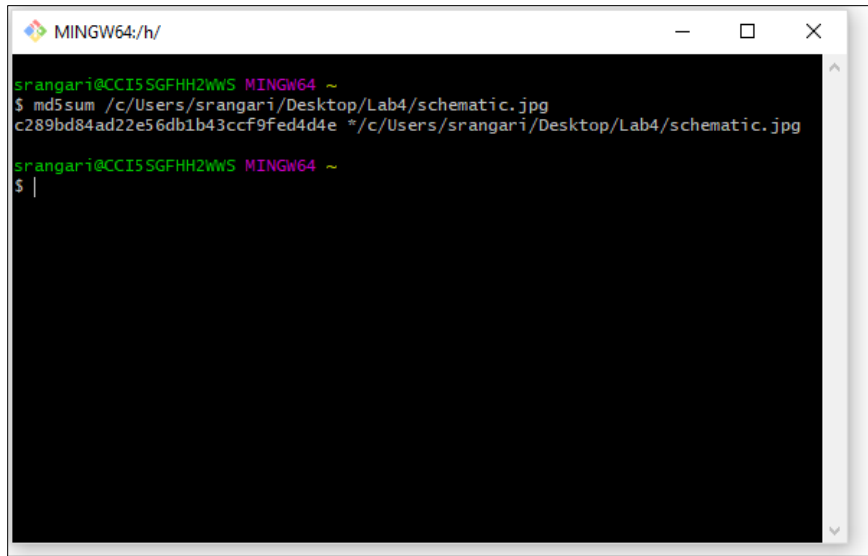
The below file shows schematic.jpg.



The hash value of the schematic:

In order to find the hash value of this schematic, I used **git bash** which is a msys shell included in "Git for Windows", and is a slimmed-down version of Cygwin (an old version at that), whose only purpose is to provide enough of a POSIX layer to run a bash.

After running a command **md5sum** and entering the location of the hidden message i.e schematic.jpg, I got hash value as shown below.



```
MINGW64: h/
srangari@CCI5SGFHH2WWS MINGW64 ~
$ md5sum /c/Users/srangari/Desktop/Lab4/schematic.jpg
c289bd84ad22e56db1b43ccf9fed4d4e */c/Users/srangari/Desktop/Lab4/schematic.jpg
srangari@CCI5SGFHH2WWS MINGW64 ~
$ |
```

Thus the hash value of 'schematic.jpg' is **c289bd84ad22e56db1b43ccf9fed4d4e**.

Conclusion

After obtaining and verifying the forensic image, I performed operation using FTK Imager Tool to find information which is hidden. In this lab I used SilentEye Tool which is a steganography tool and used to decode a file and provides a hidden message in that file. Thus using SilentEye I found hidden message of schematic.jpg in carrots.jpeg picture. Also using the GitBash, I extracted the hash value of that hidden message.

The information I found was as follows:

- 1) Total 14 duplicate files were hidden.
- 2) The steganography tool used is SilentEye
- 3) Using SilentEye I got hidden message
- 4) Using GitBash I extracted the hash value c289bd84ad22e56db1b43ccf9fed4d4e of hidden message.

Thus I examined the provided image and found out hidden information relating to the recovery of the schematic to show that someone is stealing our technology. Also I found hash value of the schematic.