Joshua Osborne

CYEN 401 001

Here is the process I used to extract the files:

1) I installed a hex editor (ghex) and downloaded some sample files to look at headers and footers of JPGs, PDFs, PNGs, and GIFs. I looked at them in ASCII and in hex format.

2) I downloaded the corrupted file and decoded it from base64 and performed a cursory examination of the ASCII data. Here, I noticed some clues, the most important being that the file contained a Microsoft Word document that was terminated by the end of the file.

3) I found the ends of the files in the ASCII and converted them to hex (i.e. %%EOF for the PDF, docProps…. For the word document, etc.). These would be the footers for my code to search for when finding the end of the file. I used some knowledge from the examination of sample files to aid my search, also searching through the hex at some times.

4) The Python program was written and revised to write each file.

Here is the description of the program:

1) Program defines class called “DocInfo.” Several objects are instantiated with deduced file header, footer, and type. Each object opens a file to write the carved information to. The class contains a function called “carve.”

2) “Carve” uses the information in the current “DocInfo” object and searches for the file in the given corrupted file. After finding the index of the header and the footer, the resulting file is written and checked using “checkHash.”

3) “checkHash” compares the md5 hash of the extracted file contents to the list of hashes given in the assignment and verifies the hash is within that list.

4) Magic numbers are defined, DocInfo objects are created and added to a list.

5) Corrupted file is converted from Base64 to ASCII to hex

6) Hex of corrupted file is put into argument of carve() function for each DocInfo object.

I used this program to extract the files and view them in my system.

Run this like any other python program (python2 file\_carving.py). Ensure the corrupted file is in the directory of the program and is named “corrupted.docx”. 5 files will be written to the system directory, and the hash for each file will be printed to stdout along with a short message. The DOC file is not in the hash list. Each file can be opened after executing the program.