A close up of a sign

Description automatically generated

Project 2

Team Member Name(s):

Rachel Sowada, Haden Stuart, and Lucia Bajo

Date 12/04/2020

# Executive Summary

A forensics analyst is provided a copy of a recovered disk image, however, they are not familiar with the file system on it. Knowing that each file on this disk image is stored in a consecutive piece of memory, the analyst can begin to pull files off of the disk.

You can determine what type of file something is by figuring out the file headers and footers. The FileRecover.py script went through the disk memory looking for the first header and then checked to see if a footer was present for the same type of file. This information was used to determine the length of the file. This began the file recovery process since the starting location and length are all that is needed in order to do so.

False positives can occur since signatures are normal bytes of data that are stored within the memory. Random data from different files can hold the same bytes therefore resulting in a false positive. There are also situations where PDF footers can be found throughout the document and in this instance the analyst must make sure they are grabbing the final signature for the entire file. The FileRecovery script took care of this.

A total of 13 files were recovered off of the disk image.

# Table of Contents

Contents

[Executive Summary 2](#_heading=h.gjdgxs)

[Table of Contents 3](#_heading=h.1fob9te)

[List of Tables 4](#_heading=h.3znysh7)

[1.0](#_heading=h.2et92p0) Problem Description 5

[2.0](#_heading=h.tyjcwt) Methodology (Analysis Techniques) 5

[2.1](#_heading=h.3dy6vkm) File Headers and Footers 5

[2.2](#_heading=h.2s8eyo1) File Size 7

[2.3](#_heading=h.17dp8vu) False Positives 7

[3.0](#_heading=h.3rdcrjn) Results (Tables and Screenshots) 7

[4.0](#_heading=h.lnxbz9) Conclusion and Recommendations 8

[7](#_heading=h.35nkun2) References 8

# List of Tables

[Table 1: Header Signatures Used to Identify Files within the Disk Image. 5](#_heading=h.1t3h5sf)

[Table 2: Footer Signatures used to Identify the End of Specific Files 6](#_heading=h.4d34og8)

[Table 3: Recovered Files from the System 8](#_heading=h.26in1rg)

# Problem Description

A copy of a recovered disk image is provided, but it is not a file system that the forensics analyst is familiar with. Using the information that each file is stored in one consecutive chunk of memory, the analyst is able to pull all of the files off of the disk,

# Methodology (Analysis Techniques)

The exact code used to recover all of the files from the disk can be found alongside this report under the name FileRecovery.py and must be run using python3. The python script used file header signatures along with footer signatures to identify and retrieve each image.

# File Headers and Footers

File headers contain unique byte strings to identify what type of file it is. File signatures can vary in length and change with different versions of a released product, but the ones used during recovery on this disk image are found in Table 1.

*Table 1: Header Signatures Used to Identify Files within the Disk Image.*

|  |  |
| --- | --- |
| File Type | Header Signature |
| MPG | \0x00\0x00\0x01\0xB3\0x14 |
| PDF | \0x25\0x50\0x44\0x46 |
| BMP | \0x42\0x4D |
| GIF | \0x47\0x49\0x46\0x38\0x39\0x61 |
| JPG | \0xFF\0xD9 |
| DOCX | \0x50\0x4B\0x05\0x06 |
| AVI | \0x52\0x49\0x46\0x46 |
| PNG | \0x89\0x50\0x4E\0x47\0x0D\0x0A\0x1A\0x0A |

The FileRecovery.py Script contained a dictionary of each of these file headers. It then went through the entire disk memory looking for the first header in the list MPG. Once that header was identified, the script looked up to see if there was a footer identified for this type of file. In most cases there was, meaning the script would then look through the remaining memory looking for the associated footer. Once the footer was identified it was recorded and used along with the header to identify the length of the file. The starting location and the length are all that is necessary to recover the file. All the footers that were needed for this disk image can be found in Table 2.

*Table 2: Footer Signatures used to Identify the End of Specific Files*

|  |  |
| --- | --- |
| File Type | Footer Signature |
| MPG | \x00\x00\x01\xB7 |
| PDF | \x25\x25\x45\x4F\x46 |
| DOCX | \x50\x4B\x05\x06 |
| PNG | \0x49\0x45\0x4E\0x44\0xAE\0x42\0x60\0x82 |
| JPG | \0xFF\0xD9 |
| GIF | \0x00\0x00\0x3B |

Once the file is recovered an MD5 hash for that file is recorded to ensure that should the file be changed at a later date it can easily be identified. All of the information found above is then added to a tuple inside the script to be printed out at the end.

The rest of the disk image is then searched for any other files of the same type, after which the process is repeated for each identified file. Once the end of the disk image is reached, the process is repeated for each type of file previously identified.

# File Size

Not all file formats contain a file footer. Instead the file size is included in the file header. The AVI files are an example of this type of file. This means to recover the files, all that needs to be done is to identify the header, look at the specified bytes that are used to identify the size and use that information to find the end of the file. Once this is achieved the same process of finding the MD5 hash and adding the information to the found file dictionary is exactly the same as what was described in the previous section.

# False Positives

Since the signatures are normal bytes of data stored in memory, it’s completely likely that random data from another file could contain identical bytes. When this happens it is a false positive in the system. This becomes more likely the shorter the signatures are, meaning BMP and GIF had the most false positives in identifying their header or footer. For BMP we learned that bytes 2-5 after the start of the data source were reserved for 00 values. Using this we could check if a BMP result was actually a piece of real data or just a false positive that was found.

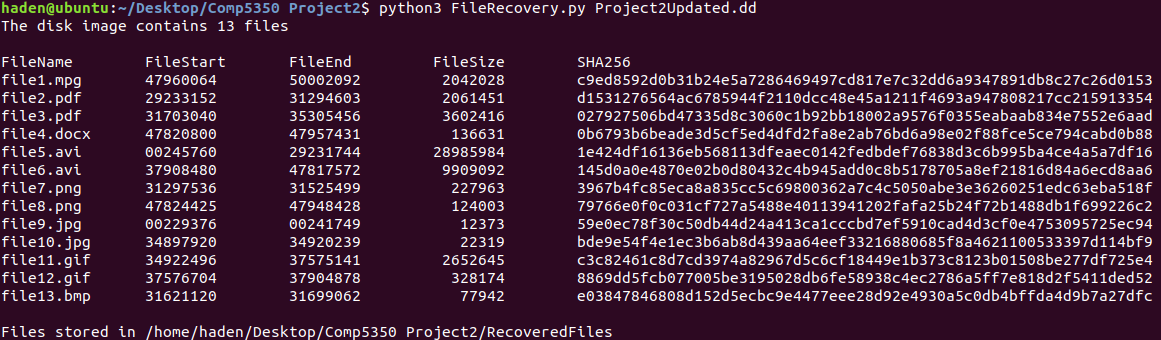
There are also cases when a signature is identified, but it is not really a false positive. A good example of this is the PDF file format. The footer of a PDF can be found throughout the document, meaning to ensure all of the data is recorded, the analyst must ensure they are grabbing the very last signature footer for that file. The way this is done in the FileRecovery script is to identify when the next PDF header is found and search for possible footers in the distance before that new file. This ensures that multiple pdf files are not accidently combined.

# Results (Tables and Screenshots)

All of the files recovered from the provided disk image can be found in Table 3. This includes not only the file type, but the starting location, ending location, file size, and the associated MD5 hash.

*Table 3: Recovered Files from the System*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| File | Starting Offset | Ending Offset | Size | SHA\_256 |
| File1.mpg | 47960064 | 50002092 | 2042028 | c9ed8592d0b31b24e5a7286469497cd817e7c32dd6a9347891db8c27c26d0153 |
| File2.pdf | 29233152 | 31294603 | 2061451 | d1531276564ac6785944f2110dcc48e45a1211f4693a947808217cc215913354 |
| File3.pdf | 31703040 | 35305456 | 3602416 | 027927506bd47335d8c3060c1b92bb18002a9576f0355eabaab834e7552e6aad |
| File4.docx | 47820800 | 47957431 | 136631 | 0b6793b6beade3d5cf5ed4dfd2fa8e2ab76bd6a98e02f88fce5ce794cabd0b88 |
| File5.avi | 00245760 | 29231744 | 28985984 | 1e424df16136eb568113dfeaec0142fedbdef76838d3c6b995ba4ce4a5a7df16 |
| File6.avi | 37908480 | 47817572 | 9909092 | 145d0a0e4870e02b0d80432c4b945add0c8b5178705a8ef21816d84a6ecd8aa6 |
| File7.png | 31297536 | 31525499 | 227963 | 3967b4fc85eca8a835cc5c69800362a7c4c5050abe3e36260251edc63eba518f |
| File8.png | 47824425 | 47948428 | 124003 | 79766e0f0c031cf727a5488e40113941202fafa25b24f72b1488db1f699226c2 |
| File9.jpg | 00229376 | 00241749 | 12373 | 59e0ec78f30c50db44d24a413ca1cccbd7ef5910cad4d3cf0e4753095725ec94 |
| File10.jpg | 34897920 | 34920239 | 22319 | bde9e54f4e1ec3b6ab8d439aa64eef33216880685f8a4621100533397d114bf9 |
| File11.gif | 34922496 | 37575141 | 2652645 | c3c82461c8d7cd3974a82967d5c6cf18449e1b373c8123b01508be277df725e4 |
| File12.gif | 37576704 | 37904878 | 328174 | 8869dd5fcb077005be3195028db6fe58938c4ec2786a5ff7e818d2f5411ded52 |
| File13.bmp | 31621120 | 31699062 | 77942 | e03847846808d152d5ecbc9e4477eee28d92e4930a5c0db4bffda4d9b7a27dfc |



# Conclusion

A total of 13 files were recovered from the disk image using the FileRecovery.py script which used file header signatures along with footer signatures to identify and retrieve each image.

# 7 References

[1] Kessler, Gary C. c2020. GCK’s File Signatures Table [Internet]. Available from: www.garykessler.net/library/file\_sigs.html