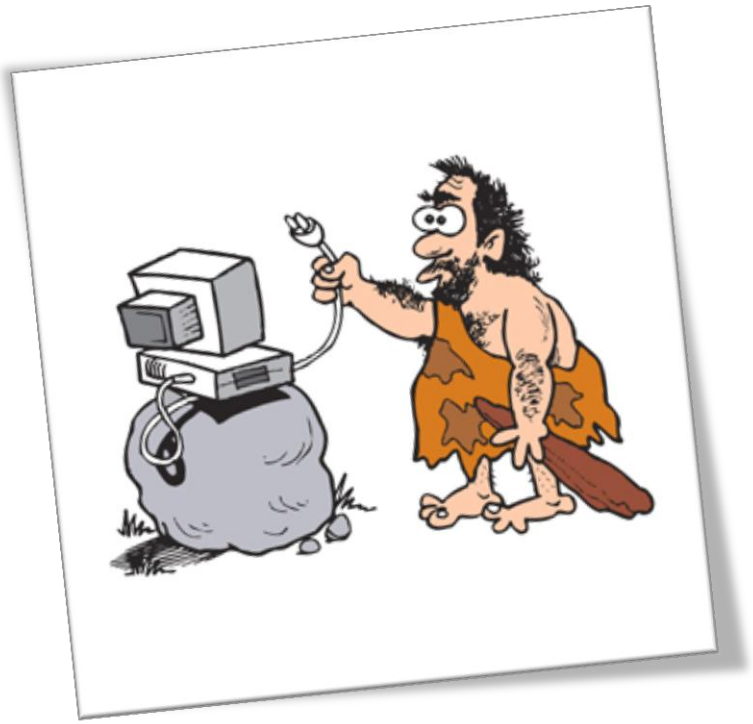
Introduction to File Carving



Introduction

The news sites are regularly reporting about the fact that confidential or secret information was compromised. The loss of an USB-stick or device from any kind of government agency or financial institute is happening quite frequently. Most of the time, the information was present on the device, but what if the information was deleted or even better, the device was formatted? After deletion, formatting and/or repartitioning we can use a technique called ‘Carving’.

‘File Carving‘ or sometimes simply carving, is the process of extracting a collection of data from a larger data set. Data carving techniques frequently occur during a digital investigation when the unallocated file system space is analyzed to extract files. The files are "carved" from the unallocated space using file type-specific header and footer values.

File system structures are not used during the process. File carving is a powerful tool for recovering files and fragments of files when directory entries are corrupt or missing. Carving is also especially useful in criminal cases, where the use of carving techniques can recover evidence. In certain cases related to child pornography, Law Enforcement agents were able to recover more images from the suspect’s hard-disks by using carving techniques.

Memory carving is a useful technique for analyzing physical and virtual memory dumps when the memory structures are unknown or have been overwritten. An example of memory dump carving is the recovery of files from a mobile phone.

In this article some basics and tooling of carving files will be explained.

File recovery vs. Carving

When data is lost on a medium, people want to recover it. There is a big difference between file recovery techniques and carving. File recovery techniques make use of the file system information that remains after deletion of a file. By using this information, many files can be recovered. A disadvantage is that the file system information needs to be correct. If not, the files can’t be recovered. If a system is formatted, the file recovery techniques will not work either. Carving works with the raw data and doesn’t use the file system structure during its process. A File system is a structure for storing and organizing computer files and the data they contain. Examples of often used file systems are: FAT16, FAT32, NTFS, EXT etc. Although carving doesn’t care about which file system is used to store the files, it could be very helpful to understand how the specific file system works. A detailed and basic book for forensics of file systems is Brain Carrier’s ‘File System Forensic Analysis’.

But how does carving work? Carving makes use of the internal structure of a file. A file is a block of stored information like an image in a jpeg file. A computer is using extensions in file names to identify what these files contain. Let’s have a look of the internal structure of a ‘jpeg’ file.

Short Name	Bytes	Payload	Name
SOI	0x FF D8	*none*	Start Of Image
SOF0	0x FF C0	*variable size*	Start Of Frame (Baseline DCT)
SOF2	0x FF C2	*variable size*	Start Of Frame (Progressive DCT)
DHT	0x FF C4	*variable size*	Define Huffman Table(s)
DQT	0x FF DB	*variable size*	Define Quantization Table(s)
DRI	0x FF DD	2 bytes	Define Restart Interval
SOS	0x FF DA	*variable size*	Start Of Stream
RST*n*	0x FF D0 … 0xFFD7	*none*	Restart
APP*n*	0x FF E*n*	*variable size*	Application-specific
COM	0x FF FE	*variable size*	Comment (text)
EOI	0x FF D9	*none*	End Of Image

Figure 1: file structure of a jpeg-file

In a jpeg file there are certain structures which could help the carving software to distinguish this type of file from the rest of the raw data. First of all , there is the header. The header is an identification string which is unique for every file type. This could be very useful to identify the beginning of file types. In our example of the JPEG file structure, the Start of Image (SOI) of a jpeg file starts with the byte values '0xFF D8' (header). Following the SOI are a series of "Marker" blocks of data used for file information. Each of these "Markers" begin with a signature "FF XX", where "XX" identifies the type of marker. The 2 bytes following each marker header is the size of the marker data. The marker data immediately follows the size and then the next marker header "FF XX" immediately follows the previous marker data. There is no standard as to how many markers will exist, but following the markers, the signature "FF DA" indicates the "Start of Stream" marker. The SOS marker is followed by a 2-byte value of the size of the SOS data and is immediately followed by the Image stream that makes up the graphic. A jpeg file ends with the bytes '0xFF D9'(footer). The constant values '0xFF D8' and '0x FF D9' are also called the 'magic numbers'

In some cases it is possible that a thumbnail graphic exists within the file. The thumbnail graphic will have the exact same components as the full-size graphic, starting with the byte values 'FF D8' and ending with the byte values of 'FF D9'. A thumbnail graphic is smaller and less likely to experience fragmentation than it's larger parent full-size graphic. Thumbnail graphics can be used as a comparison tool for evaluating what the entire jpeg graphic is to look like, in the event you must do a manual visual review of the carved graphic.

As an example, a jpeg image was viewed into Pspad Hex. In the following Figures the header and footer of the jpeg file will be shown:

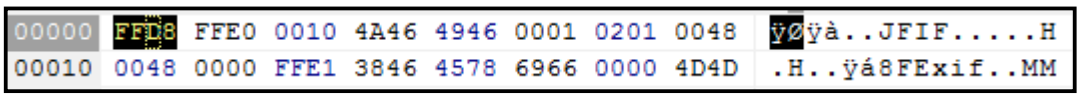


Figure 2 jpeg header

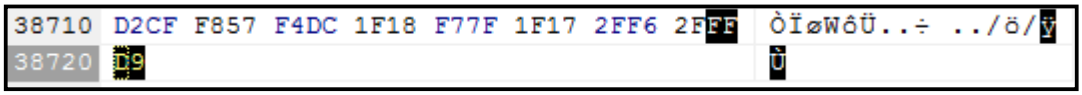


Figure 3 jpeg footer

Header-footer carving is one of the simplest ways of carving. It searches through the raw data for the file the file types you wish to carve. This kind of carving assumes that:

- The files searched for are not fragmented;
- The beginning of the file is still present;
- The signature being searched for is not a common string which could cause numerous ‘false positives’.

An example of a common string is the header of an mp3 file. This header starts with the letters ‘mp’. This is unique regarding other file types, but the signature ‘mp’ could be in many places in the raw data and is not necessary pointing to the beginning of an mp3 file.

Fragmentation:

Fragmentation appears to be relatively rare on today’s file systems. The modern operating systems try to write files without fragmentation because these files are faster to write and to read. But there are three conditions under which an operating system must write a file with two or more fragments:

1. There may be no contiguous region of sectors on the media large enough to hold the file without fragmentation. This is likely if a drive has been in use a long time, is filled near capacity, and has had many files added and deleted in more-or-less random order over time.

2. If data are appended to an existing file, there may not be sufficient unallocated sectors at the end of the file to accommodate the new data. In this case some file systems may relocate the original file, but most will simply write the appended data to another location.

3. The file system itself may not support writing files of a certain size in a contiguous manner. For example, the Unix File System will fragment files that are long or have bytes at the end of the file that will not fit into an even number of sectors.

For carving files which are fragmented, have no beginning or have a common string, we need to use some advanced carving techniques which are looking towards the file structure. This is called ‘file structure carving’. File structure carving makes use of recognizable structures outside the header and footer signatures.

Tooling
There are different carving tools available, most of them are open-source and others are commercial solutions offered by companies. Due to the fact that carving is a developing technique, more and more tools are becoming available. Some of the most common used carving tools are:

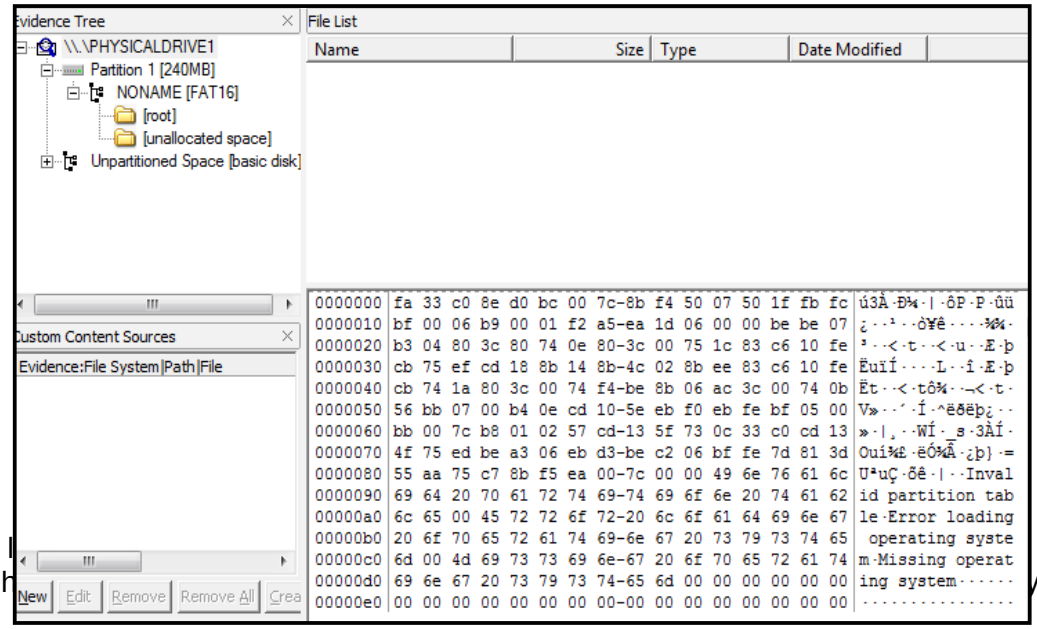
- ***Foremost***
  Originally designed by the U.S. Airforce, is a carver designed for recovering files based on their headers, footers, and internal data structures

- ***Scalpel***
  Scalpel is a rewrite of Foremost focused on performance and a decrease of memory usage. It uses a database of header and footer definitions and extracts matching files from a set of image files or raw device files. Scalpel is file system-independent and will carve files from FATx, NTFS, ext2/3, or raw partitions. Scalpel wil not allow you to output to the same directory you're carving from.

- ***Photorec***
  Photorec is a data recovery software tool designed to recover lost files from digital camera memory (CompactFlash, Memory Stick, Secure Digital, SmartMedia, Microdrive, MMC, USB flash drives etc.), hard disks and CD-ROMs. It recovers most common photo formats, including JPEG, audio files including MP3, document formats such as Microsoft Office, PDF and HTML, and archive formats including ZIP. PhotoRec does not attempt to write to the damaged media you are about to recover from. Recovered files are instead written to the directory from where you are running PhotoRec or any other directory you choose.

More info about tools:

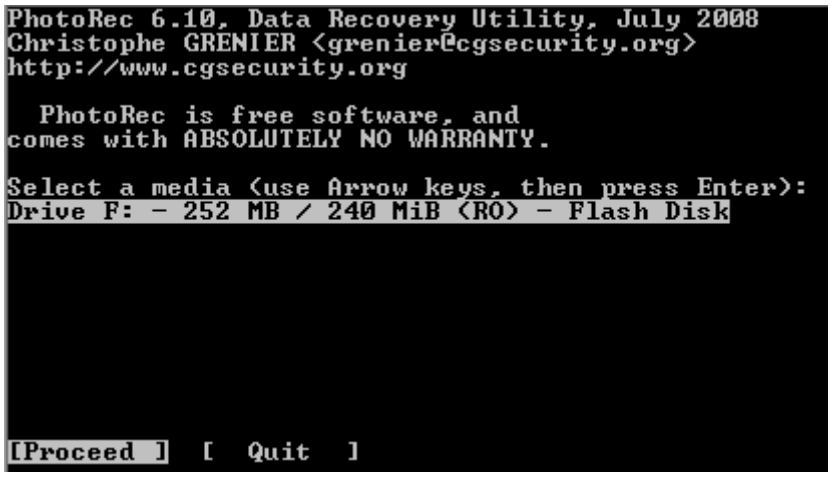
In the next part we will investigate an USB-stick which was delivered to you where the customer asks you to recover his data after he accidently formatted it. After making an image of the stick, you open the stick in FTK imager to look for any data on it:



It l
Th ygwin.

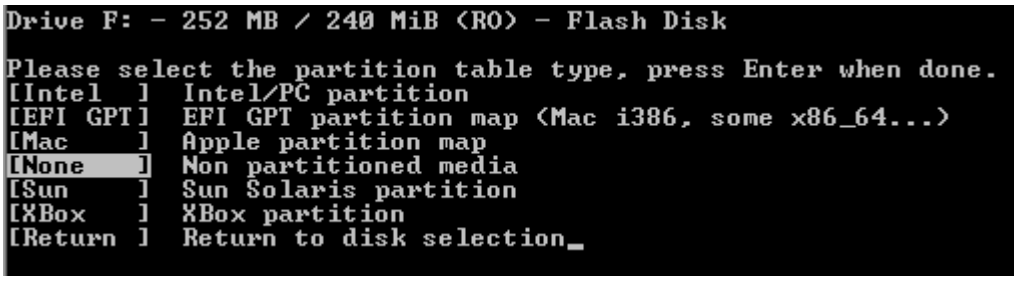
In the Cygwin command prompt, go to the location of Photorec and fire it up:

```
# ./photorec_win.exe
```



As you can see it has detected the USB-stick.

Press 'Enter' to Proceed

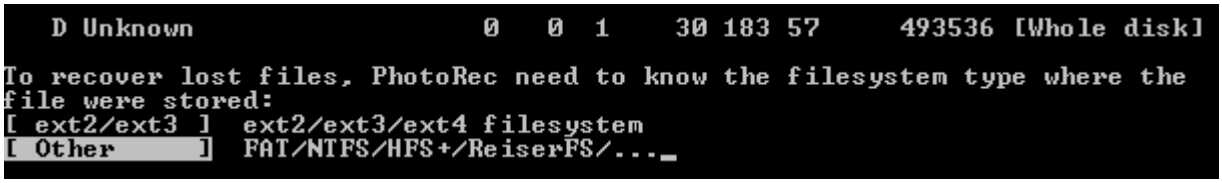


Choose the option 'None'

In the File options of Photorec it is possible to search for specific file types only. Since we don't know what's on the USB-stick, we choose to proceed to search for all file types supported by Photorec. By the way, if you miss a file type and think it's imported you can always contact the maker of Photorec, he is willing to listen to your request. Christophe

Grenier can be contacted on grenier@cgsecurity.org. For Scalpel, you can input your own header/footer combos without making contact for requesting a change in the rule config.

Choose for 'Whole disk' and then 'Search'



Choose for the option 'Other'

As we saw from FTK Imager the file system is FAT.
Photorec will ask you further where to store the carved files.

Select your destination (not on same media that you are searching) and press 'Enter'

Photorec is running and searching for the file types. It already found two headers. After a couple of minutes the scan was finished and Photorec showed the results:

Photorec carved out five files from the USB-stick. Don't forget to check if the restored files are correct. Due to fragmentation, files will not always be recovered as they should be.

To have a test image, Nick Mikus provides two test images to work with:
http://dftt.sourceforge.net/test11/index.html (Fat32 file system)
http://dftt.sourceforge.net/test12/index.html (Ext2 file system)

Mobile phones
In the previous parts we discussed carving files out of raw data and file systems. For people working in forensics, or interested in forensics, mobile phones are also very interesting sources of data. As with file systems, when you delete a file it's only deleted when it's

overwritten by other data. In FAT when a file is deleted, the file's directory entry is changed to show that the file is no longer needed. The 1st character of the filename is replaced with a 'marker', but the file data itself is left unchanged. Until it's overwritten, the data is still present.

For mobile phones it's the same. If you delete an SMS -message, it will still be in the memory of the phone until that memory space is overwritten.

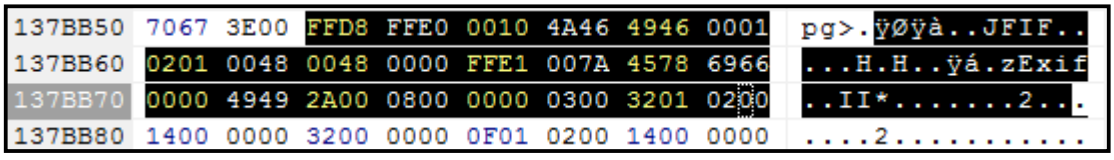
Recovering data from a mobile phone is different. All phone models have an Operating system: Windows CE, Symbian, Android, and MacOSX. These operating systems also store their files in the memory of the phone. Samsung makes use of the FAT file system. Every mobile-phone vendor has its own way for storing data into the phone memory. Some vendors store the IMSI code (subscriber identification) in a certain field in the right order, but other vendors use 'reverse nibbling' to store this code in the phone memory.

But how is it possible to recover data from a mobile phone? You need to understand the principles how the data is being stored on the mobile phone. For example the content of an SMS message is compressed by the PDU format from 8 ASCII characters into 7 bytes. There are two ways of sending and receiving SMS messages: by text mode and by PDU (protocol description unit) mode. The text mode (unavailable on some phones) is just an encoding of the bit stream represented by the PDU mode. Alphabets may differ and there are several encoding alternatives when displaying an SMS message. Photos and music are usually stored on the onboard memory card. There is no standard solution for recovering data from mobile phones.

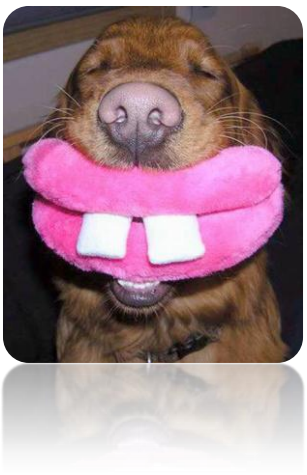
For computers, though, images of the disk and memory can be made by using the tool 'dd'. For mobile phones you need a 'flasher ' to dump the physical file system of a mobile unit. From a practitioner's point of view a 'Hex Dump' is a snapshot of the entire contents of the handsets memory. Forensic examiners are striving to grab this data, preserve it and analyze it in the hope of finding information normally hidden from view and/or deleted data. Most of the Mobile Phone forensic examination applications are a progression of 'backup software' that concentrates on the users' data. Some of the applications have the functionality to decode the data stored, but many of them do not support the recovery of deleted items.

What to do? Manually investigating the dump. Mobile phones could contain file types like: jpeg, mp3, mpeg, mov, etc. Before manually searching you need to define the file structure of each file-type. If, for example, we want to search for jpeg files in a dump from a cell phone, we could use the header and footer characteristics for jpeg. In a previous section we mentioned already these values: '0xFF D8' for the header and '0x FF D9'for the footer.

Open the dump in your favorite hex-editor and start searching for the string 'FF D8':



After finding a possible jpeg file, mark this beginning position and start looking for the values of the footer. When you have found the footer, select the block of data and save it to disk as a jpeg file. While opening it in a file viewer, the following image appears:



In this case we are lucky, the header and footer belonged to the same jpeg file. Many times you will notice that the images you retrieve by hand are incomplete. As stated before the way mobile phones store their data is random, so the files you are looking for could be heavily fragmented. Some Law Enforcement agencies have developed tools to deal with fragmented photo files, but they are not available to the public.

Summary

There can be much more said and written about data carving. This article was written to share some basic information about this technique and how it can be used easily. The development of carving tools and techniques is in its first phase and a lot has to be developed and discovered. Currently the experts are working on memory carving and recovery techniques.

References:

http://www.forensicswiki.org/wiki/File_Carving
Brain Carrier's 'File System Forensic Analysis'
S. Garfunkel 'Carving contiguous and fragmented files with fast object validation'
http://www.digital-evidence.org/
http://www.dfrws.org/

About the author

Christiaan Beek *has been working in the security field for several years. Working for national and international companies, he gained knowledge of hacking techniques, forensic analysis, scada security and incident response. Currently he is working as a security consultant/ethical hacker/security officer & trainer for a Dutch company, TenICT. He developed and lectured an internet forensics training and a digital evidence training for attorneys. As a SME he acted for the Duth News Agency on prime-time news about the TJX hack.His free time is spent with security research & writing for several media outlets. Christiaan has been invited to hold presentations and trainings in many international security conferences such as Blackhat /EU and ITUnderground.*