PyFlag: An Advanced Network Forensic Framework

By

Michael Cohen

Presented At

The Digital Forensic Research Conference

DFRWS 2008 USA Baltimore, MD (Aug 11th - 13th)

DFRWS is dedicated to the sharing of knowledge and ideas about digital forensics research. Ever since it organized the first open workshop devoted to digital forensics in 2001, DFRWS continues to bring academics and practitioners together in an informal environment. As a non-profit, volunteer organization, DFRWS sponsors technical working groups, annual conferences and challenges to help drive the direction of research and development.

http:/dfrws.org

PyFlag Forensic and Log Analysis GUI

For: DFRWS 2008

Date: 2008-08-13

Authors: Dr. Michael Cohen
Data specialist
Australian Federal Police.

Network Forensics

- When would I ever use this?
  - Sometimes during an investigation it is possible to obtain network captures.
  - System administrators might want to investigate suspicious activity by an employee for example.
  - The network captures may be of an ongoing attack.
- Legal aspects
  - I am not a lawyer!!!!!
  - There are complex legal issues regarding interception of traffic (Telcom Intercept Act, Privacy Act etc).
  - Please seek advice before you obtain the network capture.
- What do we hope to get out of this?

PyFlag Design Goals

- Manipulate large quantities of information efficiently.
    - We use a database to store and manage information.
- Perform common analysis in advance - perusal of information should be very quick.
    - Script support allows automated analysis. Caching system allows for very responsive display.
- Every inference must be directly referenced by the evidence.
        - Every detail shown must be reproducible by other tools.

IO Sources

- Often Images are supplied in a variety of formats:
    - DD Images
    - Encase Images
    - Split DD images (e.g. LogiCube)
    - RAID images
- Using an abstracted IO Source Driver we can support all those formats with the same tool.

File Systems

- File-systems are used to present and organize lots of information:
    - Users are very familiar with directory/files hierarchy
    - Many forensic tasks are related to file-system.
    - Hierarchical in nature
- File-systems use an internal representation called Inodes, and present files/directories for users.
    - PyFlag uses the Virtual File System
    - Just like a real file-system, VFS maps inodes to filenames/directory names.

VFS Internals

- Inode format:
  - Inodes are sequences of strings seperated by | (pipe)
  - Each of these strings begins with a single character referring to a registered VFS File driver. e.g.:
    - S Stream reassembler
    - P PST Driver
    - Z Zip Driver
    - G Gzip Driver

VFS Internals

- When we wish to open an inode we successively pass data from driver to driver until we get the final file:
  - Inet|S4/5|o456:30255|m1|T2
    - Means: Using the IOSource called 'net' take the combined TCP stream 4+5, at offset 456 there is a mime message, the first attachment has a tar file we want the second file in it.
    - This allows PyFlag to achieve great reach with recursive unpacking of contained files.

The FileSystem Driver

- When a FileSystem is loaded, we use a FileSystem Driver to populate the initial VFS:
  - Support many filesystems through Sleuthkit
  - PCAP Filesystem is used to process network traces
    - Uses a TCP stream reassembler written in C to creates VFS inodes for forward and reverse streams.

Scanning the VFS

- Scanners are small pieces of code which analyse files from the VFS:
    - Scanners discover new files to be inserted into the VFS.
        - e.g. ZipScanner, PSTScanner
    - Scanners can collect metadata about VFS files in external tables (not in the VFS).
        - e.g. IndexScanner, IECache Scanner
- Scanners can be recursive (i.e. Scanners will generally scan the files it discovers using all the other scanners).
- Scanners are the main way to populate the VFS.

Architecture Overview

Network Forensics

- Forensics on PCAP files is unique:
    - Most network analysis tools concentrate on the network. Provide access to packets and protocols. (e.g. Wireshark)
- Investigators typically are interested in high level details:
    - Files transferred
    - Social networks
    - Emails
    - URLs visited
    - Web Pages seen
- At the same time investigators need to pin point the packets linked with these high level events - we must always tie everything to the evidence.

Network Forensics

- PyFlag merges the Network with the standard forensic model:
  - A PCAP Filesystem driver populates the VFS with reassembled streams.
  - A set of scanners are designed to operate on PCAP Filesystem nodes.
- Network Scanners produce VFS nodes for further scanning:
  - This merges the Disk Forensic capability with the network.
  - For example, if someone has a document inside a zip file sent in an email which they downloaded over POP3 we can find it.

Packet handlers - DNS

- DNS is a valuable forensic tool
  - DNS records may have changed since the time of the investigation
  - Can usually see what domain a request was intended for.

Stream Handlers - HTTP

- HTTP is valuable from a forensic perspective
  - HTTP is much more complex than at first appears
  - Can sometimes use strings and grep but not always

HTML Rendering

- We would like to show the jury web pages as close to how they were viewed
  - Web pages are complex - include many embedded files
  - Sometimes without those files the pages looks completely different
  - Often those files are not found in the capture (cached by browser)
  - We can download them into the sundry table (Offline)

Web applications

- Web applications are a sequence of web pages
    - Each web page encodes important information in a special way (hotmail/live classic)
- Except for AJAX
    - Completely interactive - just like a real application (Gmail).

Data presentation

- Its important to be able to communicate our findings with third parties
    - We can tag inodes as important.
    - Others need to be able to view our results without special software
    - We need to produce enough information for others to verify our deductions
        - Its not about the tool - its about the evidence.

Conclusions

- PyFlag is emerging as a complete forensic solution
    - Covering Disk forensics, Network forensics and Log Analisys
    - An open and extensible framework
    - A platform for implementing cutting edge forensic techniques.

References

The wiki can be found at

- http://www.pyflag.net/

Volatility

- https://www.volatilesystems.com/default/volatility