Eric Gevaert, Fotolia

Exploring the Open Computer Forensics Architecture

GOING DUTCH

Automate the forensics process with the Dutch police department's Open Computer Forensics Architecture. **BY RALF SPENNEBERG**

Digital crime often puts the police under pressure. They don't have the staff to collect and analyze the volumes of digital evidence that often accompanies a large-scale investigation. At the same time, digital evidence is becoming increasingly important – data on mobile phones and computers belonging to suspects can provide circumstantial evidence and even hard facts. The Dutch police [1] developed the Open Computer Forensics Architecture (OCFA [2]) as an open source tool for professional criminal investigators. Dutch authorities use the modular OCFA framework for forensic investigations. The OCFA architecture is a combination of several existing forensic tools and libraries. OCFA splits the forensic process into two parts. First, specialists with knowledge of digital forensics extract content from hard disks and other devices. Then, criminal investigators use a simple web interface to analyze the data and look for evidence.

Installable OCFA 2.0.2 packages exist for Debian Etch, Ubuntu 5.10, and SUSE 9.3 and 10.1. The tarballs include OCFA RPMs or DEBs, along with a number of additional packages and installation guides, which also describe the packages you need to install manually up front. The current 2.1.0 version is available as a source package only. The creators of OCFA see the analysis process as a kind of digital data wash (Digiwash) and, therefore, install OCFA in the */usr/local/digiwash* directory.

One of the biggest obstacles to forensic analysis is the sheer bulk of evidence. Investigators face the task of identifying incriminating material among hundreds of gigabytes of irrelevant data. But skipping files and directories just because the names sound nondescript is no solution either. Many forensic tools assist the investigator by performing automatic analysis and characterization of the

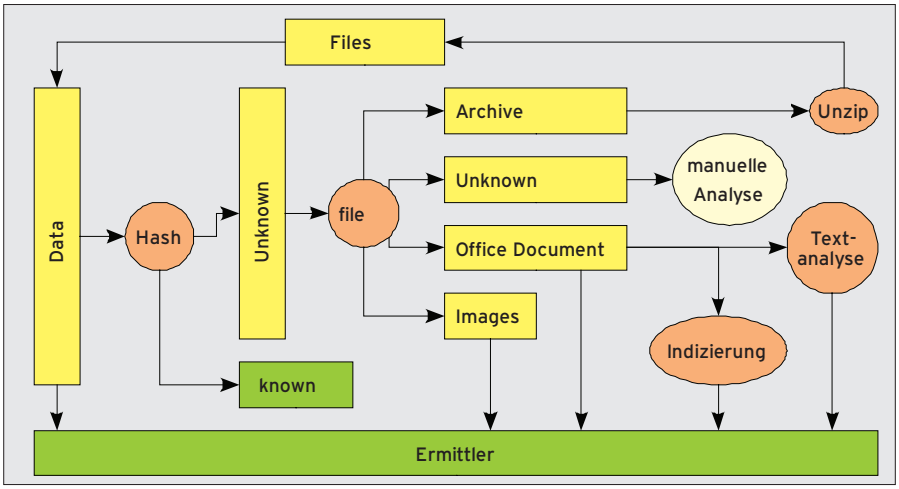

Figure 1: OCFA automatically analyzes the flood of data and places it at the disposal of investigators.

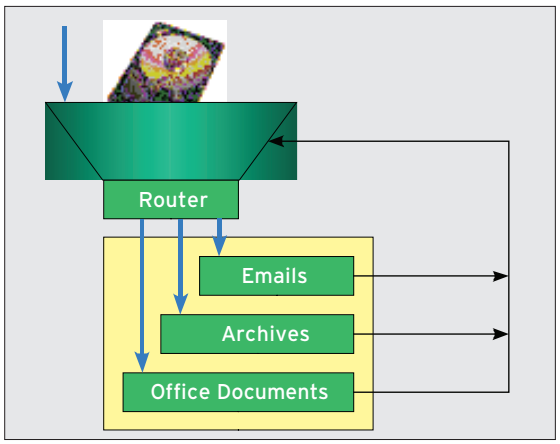

Figure 2: The router calls modules to reflect the file formats and returns any files it found to the analysis process.

identified files. Digiwash takes this idea one step further by running *file* to identify the file type. It then goes on to automatically analyze specific file types, thus saving the forensic investigator some of the grunt work. OCFA uses Lucene to index Microsoft Word files and other Office documents. The raw text is extracted by running *antiword*. PDF files are converted with *pdftotext*; *mailwash* extracts files and metadata from mailboxes. The developers have even devised a means for capturing the information in PGP keyrings, mapping the key IDs of signed and encrypted mail to clear text names. OCFA also groups photos and generates thumbnails.

The framework automatically dissects zipped containers and analyzes the files gained by this process. In this way, the framework recursively analyzes all the data and places it at the disposal of the investigators (Figure 1).

Useful Fingerprints

To reduce the volume of data for analysis, forensic investigators can integrate hash databases of known files. The databases contain MD5 or SHA1 checksums of files that investigators can safely ignore. For example, any unchanged files belonging to the operating system are bound to be irrelevant, although investigators will be interested in any modifications caused by trojans.

Databases of checksums for known files are available from the National Institute for Standards and Technology (NIST) as free downloads [3]. Other forensic tools, such as Autopsy [4], also rely on the NIST's National Software Reference Library (NSRL). The library is available in zipped format as a collection of four CD images. A Perl script in the OCFA package references the ISO files to calculate its own hash sets, which the admin then has to copy to the *digiwash* directory. The conversion process takes some time to complete.

Architecture

The router is a central part of the OCFA architecture that is responsible for recursive file processing, which it analyzes by calling external software before returning any files the process reveals to the analysis process (Figure 2). An *anycast* relay handles the communications between the individual modules. The relay coordinates messaging and also handles load balancing. This approach lets investigators run multiple instances of a module in a distributed environment on multiple computers; in other words, OCFA supports clustering.

The OCFA Framework can use additional external software packages if necessary. A patch lets users integrate the TSK [5] and Scalpel [6] forensic tools.

Figure 3: OCFA's web-based graphical interface has an antiquated charm. It doesn't offer much in the line of options, but it is quick.

Mixed Interfaces

Data recovered through OCFA is available to the forensic investigator at the command line. The investigator needs root privileges to initiate a case, because creating a new case involves restarting the Apache web server. From the web server's point of view, each case is a *VirtualHost*. The investigator then uses Apache to access the extracted data (Figure 3).

The interface also tells the investigator whether data extraction is complete. The web GUI shows the current queues and current status. In other words, OCFA actually automates communications between the investigator and the forensics expert.

The Dutch police have developed a Windows program for internal use. The program, dubbed Washbrush, analyzes Outlook and Outlook Express mailboxes and passes the results in to Digiwash. The program is only available to the Dutch authorities as of this writing. Work is also in progress on additional OCFA modules and a more state-of-art front end. The software will not be GPL'd, but it will be available via NDA (non-disclosure agreement).

Mass Processing

The complex and time-consuming OCFA installation is worthwhile in environments that support large and complex forensic analysis projects – and in cases in which the forensic and investigative tasks are easily separated. Note, however, that the sparse OCFA GUIs do not offer the convenience of other forensic tools. ■

INFO

[1] Dutch police homepage: *http://www.politie.nl/English/*
[2] OCFA: *http://ocfa.sourceforge.net*
[3] NIST NSRL: *http://www.nsrl.nist.gov*
[4] Autopsy: *http://www.sleuthkit.org/autopsy/*
[5] The Sleuth Kit: *http://www.sleuthkit.org/*
[6] Scalpel: *http://www.digitalforensicssolutions.com/Scalpel/*

THE AUTHOR

Ralf Spenneberg works as a freelance Unix/Linux trainer, consultant, and author. He has published several books on the subject of intrusion detection, firewalling, and virtual private networks. His latest book "SELinux & AppArmor" was published recently.