International Journal of Digital Crime and Forensics
Volume 12 • Issue 1 • January-March 2020

Evaluation of Autopsy and Volatility for Cybercrime Investigation

A Forensic Lucid Case Study

Ahmed Almutairi, Concordia University, Quebec, Canada
Behzad Shoarian Satari, Concordia University, Quebec, Canada
Carlos Rivas, Concordia University, Quebec, Canada
Cristian Florin Stanciu, Concordia University, Quebec, Canada
Mozhdeh Yamani, Concordia University, Quebec, Canada
Zahra Zohoorsaadat, Concordia University, Quebec, Canada
Serguei A. Mokhov, Concordia University, Quebec, Canada

ABSTRACT

In this article, the authors successfully created two new plugins one for Autopsy Forensic Tool, and the other for Volatility Framework. Both plugins are useful for encoding digital evidences in Forensic Lucid which is the goal of this work. The first plugin was integrated in Autopsy to generate a report for the case of a Brute Force Authentication attack by looking for evidence in server logs based on a key search. On the other hand, the second plugin named ForensicLucidDeviceTree aims to find whether a device stack has been infected by a root-kit or not expression is implied by the previous statement. The results of both plugins are shown in Forensic Lucid Format and were successfully compiled using GIPC compiler.

KEYWORDS

Autopsy, Forensic Lucid, GIPC Compiler, Volatility

1. INTRODUCTION

1.1. Motivation

The motivation behind this project is to re-evaluate the open source forensic tools through their hands-on use, such as that of Sleuthkit (Carrier, n.d.), and more of its Autopsy (Carrier, n.d.), and other tools in a simulated investigation, reasoning, analysis, and reporting for sample cases. The use of tools is followed by adaptation and encoding of the case's knowledge base (output) extracted from forensic artifact analysis in Forensic Lucid. Thus, the tools should be evaluated how easy is to extract their outputs, reports, and translate into the format for Forensic Lucid. The sample data would come from the honeynet (Honeynet Project, 2015) and DFRWS (Palmer, 2001) projects/challenges.

1.2. Overview

In Section 2 we provide a detailed background of our research on Autopsy (Carrier, n.d.), Volatility, Forensic Lucid, and GIPSY, that has a Forensic Lucid compiler – GIPC. In Section 3 we detail our experiments, writing plug-ins for Autopsy and Volatility, and encoding sample data output into Forensic Lucid. We conclude in Section 5.

DOI: 10.4018/IJDCF.2020010104

2. BACKGROUND

2.1. Autopsy

Autopsy is an Open Source application under the GNU General Public License Version 3. Autopsy forensics browser version 4.5.0 installs a Graphical digital forensics interface to the command line digital investigation tool Sleuth Kit and is built into the SANS investigative Forensic Toolkit workstation (SIFT Workstation) that is downloadable from https://www.autopsy.com or https://www.sleuthkit.org/autopsy/. Autopsy can be installed under Microsoft Windows 7/8.1/10, Linux Ubuntu and Mac OS X.

Autopsy tool allows to examine a hard drive or mobile device and recover evidence from it. The data sources which is supported by this tool are Disk images and VM file (e.g. Raw images (img, dd, 001, aa, raw, bin), Encase Images (E01), Virtual Machine images (vmdk, vhd formats), Local disk (local PC/server/laptop disk) and logical files (local files and directory to be added).

The ingest modules in Autopsy are responsible for the data analysis and subsequent data extraction for operations such as tagging and reporting. The tool has the following standard Ingest modules: File Type Identification, Recent Activity, and Hash database lookup, Embedded File Extractor, EXIF Parser, Email Parser, Virtual machine Extractor, and Photo Record Carver. Autopsy supports 3rd party ingest module such as Python plugins, Pre-fetch parser, Windows registry ingest modules, Virus total online checker, and Image fingerprint module.

Autopsy uses keyword search to analyze the disk image contents. The standard keyword list in autopsy includes IP addresses, phone numbers, email addresses and credit card numbers, and URLs. The group of keywords can be defined as an exact match, a substring match, or as a regular expression. The group of keywords will be created to look for a specific cyber offense such as a brute force authentication, spamming or click-thru fraud, found within an apache web server log files.

Tagging of data can be done on the files of the data source or on the search results such as keyword hits, email addresses, extracted content, interesting items, and accounts. Autopsy has two tagging categories: tagging by file (i.e. the file that includes the result) and tagging by result (e.g., keyword hits within each file). Once the items are tagged, they are added automatically into a user interface section called Tagged Results.

Autopsy offers the possibility to create a timeline for all events from the files on the disk. Furthermore, Autopsy has the capability of creating reports out of analyzed data and any tagged files. The investigator also can create a custom result by adding the tagged files showing the facts regarding certain cyber offenses. The reports in Autopsy can be generated in various formats such as HTML, Excel, CSV, Text, and tagged hashes, among others.

In this project, we have considered Autopsy version 4.5.0, However Autopsy has released version 4.6.1 on Feb. 2018. In order to gather data on some types of attacks, we used a honeypot image that is dumped from an Apache web server downloaded from apache_logs.tar.gz. There are four types of attack that can be examined on the mentioned data source:

1. Brute force Authentication: This type of attack is launched through the server by HTTP GET Requests and HTTP POST requests. It could be detected by examining the audit files.
2. IRC connection: IRC (Internet Relay Chat) is a protocol for real-time text messaging between internet-connected computers. It is mostly used for hiding the real IP address to launch denial of service attack. This type of attack could be detected by searching the audit log files for all entries to common IRC ports.
3. Spamming: A large number of users were spammers trying to send their emails through the server to hide their true location and make the tracking of the email's origin difficult.. This attack could be identified by examining the audit log files.