ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 5, February 2016 in association with
HEERA COLLEGE OF ENGINEERING AND TECHNOLOGY, THIRUVANANTHAPURAM
Organizes
NATIONAL CONFERENCE ON ENGINEERING FOR LIFE (NCEL – 2016)
(12th -13th February 2016)

ACQUISITION OF VOLATILE DATA FROM LINUX SYSTEM

Vijay N Anand
Mtech in Computer science with Specialization in Cyber Forensics & Information Technology
ER&DCI–IT
Trivandrum, Kerala, India
vijaynandakumaranand@gmail.com
Razeem Ahmad B T
Mtech in Computer science with Specialization in Cyber Forensics & Information Technology
ER&DCI–IT
Trivandrum, Kerala, India
razeem.ahmad@gmail.com

Abstract— **The Linux operating system has been used as a server system in many of business services worldwide. Nowadays, a lot of incident response approaches on such kind of platform have been established by many researchers active in the computer forensic discipline. But there is no tool to completely acquire the volatile information from a Linux based system. Thus, this paper aims to develop and implement a new framework to deal with a compromised Linux system in a live forensic investigation. The acquisition process collects open files, open ports, running process, logs and other artifacts that changes during the system restart. The aim is to develop a tool which acquires volatile artifacts from proc directory and RAM memory from a Linux system.**

Keywords—proc; kernel; kmem

I. INTRODUCTION

Live Analysis is the simplest technique through which investigator interrelate with OS such as command shell or a telnet connection. Through this technique, the investigator can get the vital system information such as logged on users, open ports, network connections, list of processes etc [1]. Live analysis might not generate reliable outcome but it is helpful in many cases. Forensic live analysis and event reconstruction methods in digital crime investigation is forensically interesting because it helps to determine the origin of events by gathering data for analysis and applying the methods of event reconstruction for evidential purposes in the court of law [3]. Linux is a Unix-like and mostly POSIX-compliant computer operating system (OS) assembled under the model of free and open-source software development and distribution. The defining component of Linux is the Linux kernel, an operating system kernel first released on 5 October 1991 by Linus Torvalds. The Free Software Foundation uses the name GNU/Linux to describe the operating system, which has led to some controversy. Linux was originally developed as a free operating system for personal computers based on the Intel x86 architecture, but has since been ported to more computer hardware platforms than any other operating system. In earlier versions of Linux, acquisition was done using dd tool. But the newer versions has brought a separate location to store a part of the RAM memory. The location is pointed by dev/kmem directory. This location is accessible only by using a kernel program or a kernel module. Other related data can be accessed using shell scripts and outputting the values or results into an xml file. So this paper focus on how to develop a kernel module that can easily collect the whole RAM data from any versions of Linux kernel.

The contents of RAM could reveal malicious code running on the system that has been deleted from the hard drive or, better yet, that was never resident on the hard drive at all. RAM can also provide the programs most recently run and files most recently opened in the system [5]. However, due to the nature of modern operating systems, these programs and files are not typically stored contiguously—which makes most retrieval efforts of files larger than one page size futile. To date, analysis of RAM images has been largely restricted to searching for ASCII string content, which typically only yields text information such as document fragments, passwords or scripts.

Other than RAM volatile data in Linux include logs, process information, logged on users, and other details present in the /proc directory of Linux memory. The /proc contains details about the running processes, network connections and other details in the running system.

II. CURRENT PRACTICES

A. *Dd tool*

Dd is a command in Linux which is used for acquisition of different files. These files may also include the RAM memory. But using dd in later linux version is not possible due to the separation and user restriction in RAM storage location.

B. *Using Firewire device*

Firewire technology was first introduced by Apple. It provided the user faster access to memory using DMA (Direct Memory Access). This technology can be used for acquisition of RAM comparatively faster than any other method. But this requires a host machine loaded with Linux to perform acquisition. It can also result in blue screen issues in victim system, since acquisition is done while system is running.

III. SYSTEM DESIGN

This paper proposes a system which extracts the complete dump of volatile data present in both RAM and other volatile memory locations.

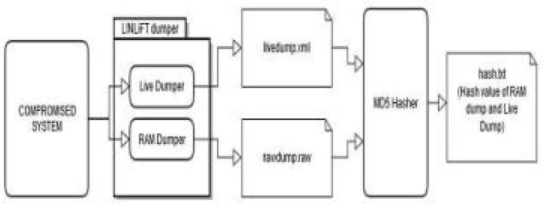

Fig 1: Design of Proposed System

Fig 1 represents the design of proposed system .It consist of two sub parts live Dumper and a RAM Dumper. Output live dumper is an xml file and that of RAM dumper is in raw format. Then an md5 hash is obtained using an md5 hasher to preserve the integrity of evidence collected.

Volatile data acquisition module

This module acquires the volatile data directly from a running system. The output of this module is formatted in .xml. This is done to enhance the working of analysis part which can be further developed. Xml format is supported in many of the analysis tools. The output of this module contains the logs, running process, network connections etc.

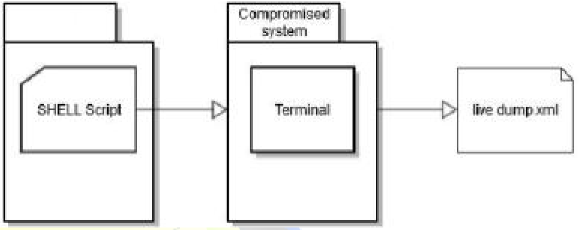

Fig 2: Volatile data Acquisition Module

Fig 2 represents the module which is used for the above purpose. The module runs a shell script in the terminal of the compromised system. Various shell commands are present in Linux for extraction of volatile data [6]. These commands are included in the shell script used in this module. The result of the shell script is piped into an xml file.

RAM memory acquisition module

The main step in acquisition process is getting access to the physical memory. For this purpose a device driver module need to be developed. This is done because in kernel mode access is required. From kernel version 2.6 and above the access to dev/mem is not available from user level. This is because of the separation RAM memory in dev/mem and dev/kmem. This module contains a user level program and a kernel level program. The user level program injects a kernel module into the kernel level of the system and install it. The kernel module creates the dump of physical memory and transfers it back to the user level program from where user can access it easily.

This helps in dumping the complete physical memory of a Linux system.

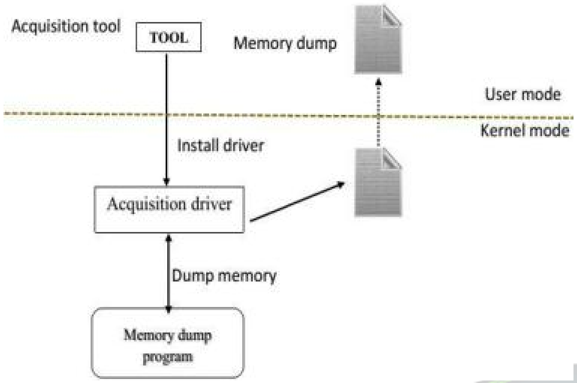

Fig 3: RAM acquisition module

IV. RELATED WORKS

There are similar works in this field. Some of them are described below.

LIME

A Loadable Kernel Module (LKM) which allows for volatile memory acquisition from Linux and Linux-based devices, such as Android. This makes LIME unique as it is the first tool that allows for full memory captures on Android devices. It also minimizes its interaction between user and kernel space processes during acquisition, which allows it to produce memory captures that are more forensically sound than those of other tools designed for Linux memory acquisition.

Lime concentrates only on RAM. It does not dump volatile memory in other locations.

Draugr

Draugr is a tool in python which can find kernel symbols, process and can be used to dis assemble the memory dump. This tool does not focus on complete dumping of physical memory.

Memfetch

Memfetch is a utility which mainly concentrates on the running process memory. This module is so lite that it cannot acquire the complete RAM dump and other volatile data present in other locations are not handled.

V. CONCLUSION

Acquisition of volatile data is an elementary part in live forensics. Rudimentary tools such as dd tool and Firewire devices are not sufficient to be applied in latest Linux systems with new kernels. Also related works does not meet the requirements. So this paper describes a method by which we can completely dump the volatile data.

VI. FUTURE SCOPE

This work can be later enhanced using efficient methods in acquisition. The proposed system retains much foot prints in compromised system, which is not forensically sound. File system analysis can be done in future instead of shell scripts, which provides a much efficient acquisition without affecting system. Furthermore proposed system's RAM dump module requires root privilege. If a system that can reside in RAM and can collect the RAM details will be easier than that of the proposed system. Such system will not leave traces in compromised system and it will be a small sized module

REFERENCES

[1] Hay et al. in, "Live Analysis: Progress and Challenges," (2009)

[2] B. D. Carrier, "Risks of live digital forensic analysis," (2006)

[3] In the paper "Forensic Live Response and Event Reconstruction Methods in Linux Systems" by Funminiyi Olajide et al.R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[4] Bard Egil Hermansen in "An Evaluation of Forensic Tools for Linux" (2010)

[5] Jorge Mario Urrea in "An analysis of linux ram forensics" (2006)

[6] http://www.malwarefieldguide.com/LinuxChapter1.html.