An Evidence Collection and Analysis of Ubuntu File System

Dinesh N. Patil[1], Bandu B. Meshram[2]
Veermata Jijabai Technological Institute
Matunga, Mumbai, India
dinesh9371@gmail.com[1], bbmeshram@vjti.org.in[2]

ABSTRACT

A file system of Ubuntu operating system can conserve and manage a lot of configuration information and the information with forensic importance. Mining and analyzing the useful data of the Ubuntu operating system have become essential with the rise of the attack on the computer system. Investigating the File System can help to collect information relevant to the case. After considering existing research and tools, this paper suggests a new evidence collection and analysis methodology and the UbuntuForensic tool to aid in the process of digital forensic investigation of Ubuntu File System. The paper also discusses a technique for the identification of the files modified by the criminal.

KEYWORDS

File System, Digital Forensic, Integrated Analysis, Timeline Analysis, Digital Evidence

1 INTRODUCTION

The Ubuntu operating system is one of the distributions of the Linux operating system. Most of the Ubuntu kernels are the default Linux kernel. Ubuntu uses the Linux file system which is usually considered as a tree structure. Ubuntu is having Ext4 as its default file system. Ext4 is an evolution of Ext3, which was the default file system earlier. Linux computers are very much prone to attack from the hackers. Linux boxes are often used as servers, essentially for a central control point. In fact, roughly 70% of malware downloaded by hackers to the honeypots is infected with Linux/Rst-B [1]. Linux-based web servers are constantly under attack. At SophosLabs, an average of 16,000-24,000 websites were compromised in a day in 2013 [2]. Linux systems are indeed attacked by malware.

The Microsoft's operating system design includes some features that make documents able to install executable payloads. The use of a database of software hooks and code stubs (the registry) also simplified things [3]. Linux malware is quite distinct from what it does and how it does it, compared to Windows viruses, but it exists. The crucial operating system directories might be used by the malware to affect the computer system as a whole. In addition, there is always the risk of the malicious insider. Attacks directed at Linux systems tend to aim at exploiting bugs in system services such as web browsers or Java containers. These don't frequently run with elevated privileges either, so an exploit is typically contained to altering the behavior of the targeted service and, possibly, disabling it. The malware uses the various directories in the Linux file system to plant it to run as a service and harm the Computer. Also, the activity of the malicious insider also gets stored in the file system. This raises the need to do the forensic investigation of directories under the Linux file system to find the traces of malicious activities on the system.

The paper is organized as follows: Section 2 discusses the related work and the existing tools on the Linux file system forensics. Section 3 covers the forensic investigation of the various user activities on the Linux file system. The proposed UbuntuForensic tool is discussed in section 4. Comparative study between the existing Linux tools and the proposed tool is

performed in section 5. The findings are concluded in section 6.

2 RELATED RESEARCH

This section details out the existing research on the Linux file system forensic and the tool developed to carry out the forensic investigation of it.

2.1 Existing Research

The logging system is the most important mechanism for Computer forensics on an Operating System. The various logging mechanism in Linux system that can be of forensic importance is discussed in [4]. A comparative study of the various file systems in Ubuntu Linux and Free BSD is performed in [5]. In order to meet the Linux file system analysis applications demand for computer forensics, an object-oriented method of analyzing Linux file system is proposed in [6]. The paper also analyzed different data sources deeply with the inheritance relationship between classes and the encapsulation of class, and showed information of Linux file to the users in a friendly interface. The Linux operating system has been used as a server system in plenty of business services worldwide. Unauthorized intrusions on a server are constantly increasing with a geometric progression. Conversely, the protection and prevention techniques against intrusion accidents are certainly insufficient. A new framework to deal with a compromised Linux system in a digital forensic investigation is developed and implemented in [7]. Issues pertaining to the Linux Forensics and the various forensic tools for the forensic investigation of the Linux system have been discussed in [8].

2.2 Existing Tools

The Sleuth kit(TSK)

It is a collection of Unix-based command line analysis tools. TSK can analyze FAT, NTFS, Ext2/3, and UFS file systems and can list files and directories, recover deleted files, make timelines of file activity, perform keyword searches, and use hash databases.

Autopsy

This tool is a graphical interface to the TSK. It also analyzes FAT, NTFS, Ext2/3, and UFS file systems and can list files and directories,
recover deleted files, make timelines of file activity, perform keyword searches, and use hash databases.

Scalpel

Scalpel is an open source file carver which is also available for Linux. File carvers are used to recover data from disks and to retrieve files from raw disk images. In some case, file carvers are even able to retrieve data if the metadata of the file system were destroyed. Scalpel is designed to use minimal resources and to perform file carving.

Digital Evidence and Forensic Toolkit (DEFT) Linux

DEFT is a free computer forensics Linux distribution. DEFT is combined with the Digital Advanced Response Toolkit (DART) which contains a collection of forensics software for Windows.

Computer Aided Investigative Environment (CAINE)

CAINE is a Linux live distribution which aims to provide a collection of forensics tools with a GUI. It includes open source tools that support the investigator in four phases of the forensic process viz., Information gathering, collection, examination, analysis. It also supports the investigator by providing capabilities to automate the creation of the final report and is completely controlled by a GUI that is organized according to the forensics phases.

i-Nex

It is an application that gathers information for hardware components available on the system and displays using user interface [9].

History

he history command lists commands that were recently executed. This can help to track the activity of an intruder.

3 EVIDENCE COLLECTION USING PROPOSED TOOL

The forensic investigator should be able to analyze the activities of the user when performing the investigation and in doing so the timing of the activities is needed to be considered to establish the correlation between the time and the activity. As the details of the user's activities are recorded in the various files managed by the file system of the Linux based Computer System. The investigator should be able to investigate the files stored in the seized hard disk of the computer system which was used to commit the crime.

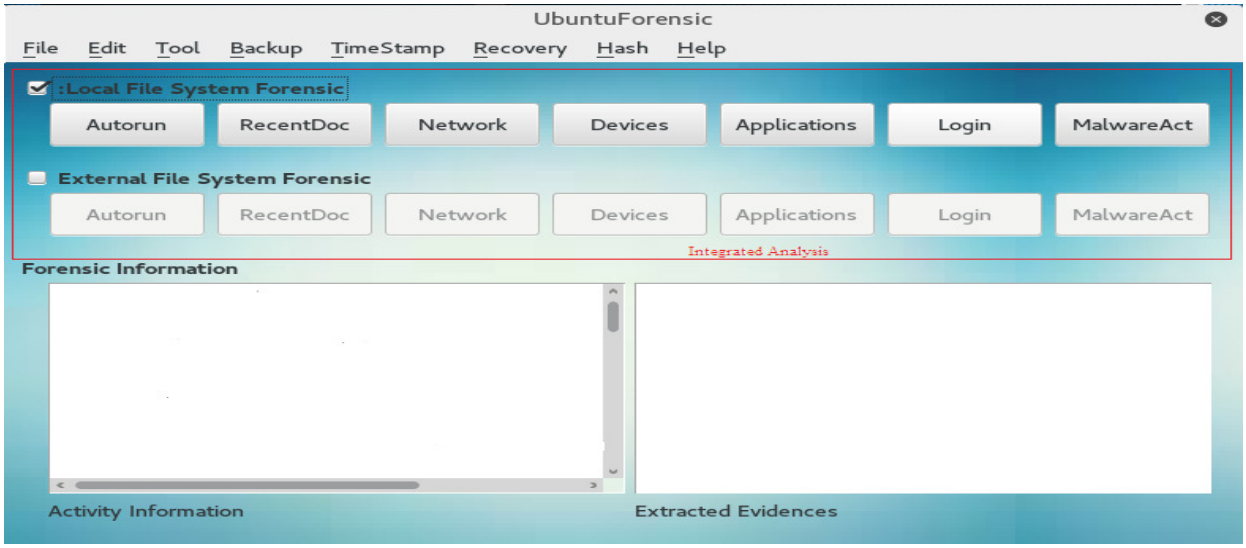

Figure 1. A snapshot of UbuntuForensic tool showing Integrated Analysis

However, the previous forensic tools provided limited facilities for performing the forensic analysis of Linux file system. For this reason, a new evidence collection and analysis methodology is required. This methodology performs integrated file system analysis, timeline analysis and extracts the information that is useful for the digital forensic analysis of the file system.

3.1 Integrated Analysis

The cyber crime cell generally used to seize the hard disk of the computer which is used for crime purpose. The forensic investigator has the responsibility to find out the possible traces of evidence against the criminal. The Linux-based computer system maintains the files in the directory structure which begin with root directory '/'.

The proposed UbuntuForensic tool provides the facility for extracting the forensic evidence from the files stored in the external hard disk. This hard disk is needed to be connected to the computer system having an UbuntuForensic tool which mounts the external directory structure in the media directory of the running system to extract the evidence. The proposed tool also performs Local file system forensic which involves extracting the information from the files about the various activity performed by the user on the system, on which the tool is running.

3.2 Analysis of User Activity

The existing tools provide a limited functionality in extracting the forensic information from the file system. This has stimulated the need of having a file system forensic tool which can extract the forensic data from the directory structure based on the various activities being performed by the user and generate a report of the evidence for further use.

The proposed UbuntuForensic tool covers the various activities as discussed in [10], which are performed on the Computer system. These activities include:

- Autorun programs running on the system
- Recently accessed documents/programs,
- Applications installed on the system
- Network connected
- Devices connected to the system
- Last login activity of the user
- Malware activity

The detail of these activities is as follows:

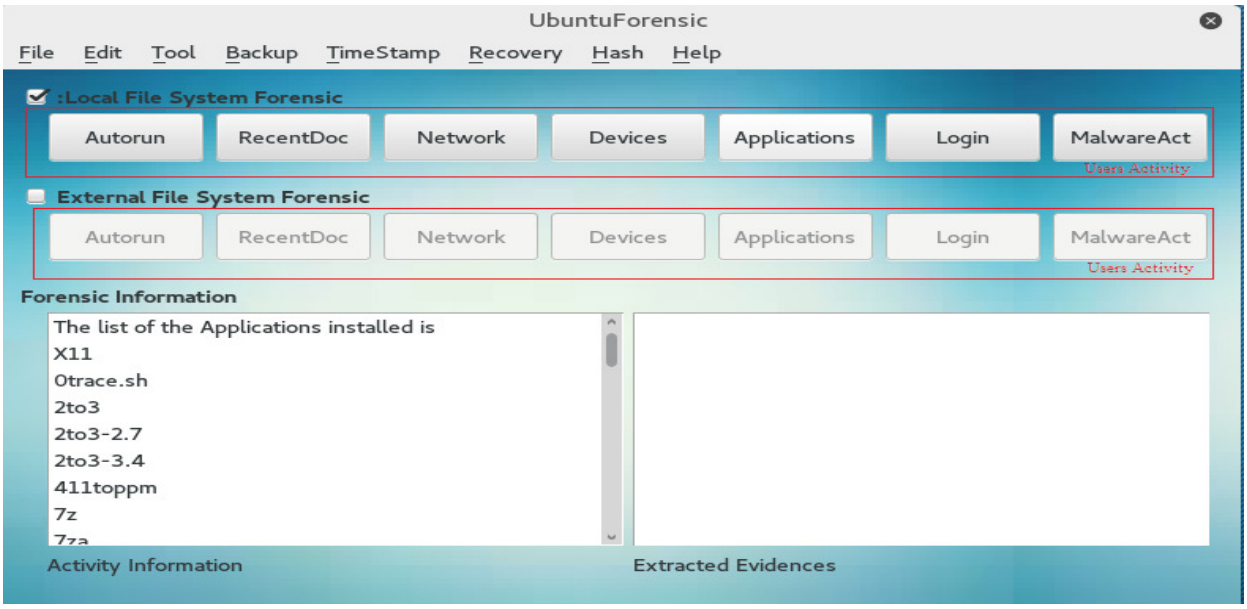

Figure 2. A snapshot of UbuntuForensic tool showing category of User Activities

The Autorun programs running on the system

Many programs are configured in such a way that when the Computer boot and start the operating system, they automatically start running such programs are called as Auto Run program. In the case of Ubuntu, the information about the programs which are to be executed when the system booted is available in the file stored /etc/rc.d directory. The malicious user might gain an access to the Ubuntu system & will add files in rc.d. So whenever the Ubuntu System will boot up the malicious script will automatically run. The forensic examiner will have to look into those files to identify if any file contains malicious code which may be causing unauthorized activity on the system.

Recently Accessed documents and programs

From the documents that the user has recently accessed, the forensic examiner can know about the documents in which the user has interest. In Ubuntu, the files which have been recently accessed are noted in the file 'recently-used.xbel'. This file is available in the local/share/ directory. The 'cat' command can be used to read the contents of the recently-used.xbel file. Recently-used.xbel file provides the detailed information about the files which have been accessed by the user, the application used to access those documents and the timing of accessing & modifying these documents.

The recently accessed document information helps in understanding the files which may have been read, modified by the user.

ISBN: 978-1-941968-37-6 ©2016 SDIWC

Applications installed on the system

In Ubuntu, the configuration information about the application is stored in the /usr/bin directory and the library required for these applications is available in the /usr/lib directory. The list of the application installed can be obtained by the command ls –l /usr/bin/. Using the information available in the bin directory, analyst can provide the historic view of the application configuration that the user has installed onto the system, date on which a particular application was modified, permissions granted to the user, size of the application etc.

Network connected or accessed

Ubuntu maintains the list of networks connected to the system in /etc/NetworkManager/system-connections. In addition to this, it is possible to know the active network connections which are being used in the system using the command "sudo netstat –tupn".

Syslog file in /var/log provides the date and time at which a particular network connection was established. Network information enables the forensic examiner to know about the type of network used in order to do malicious activity.

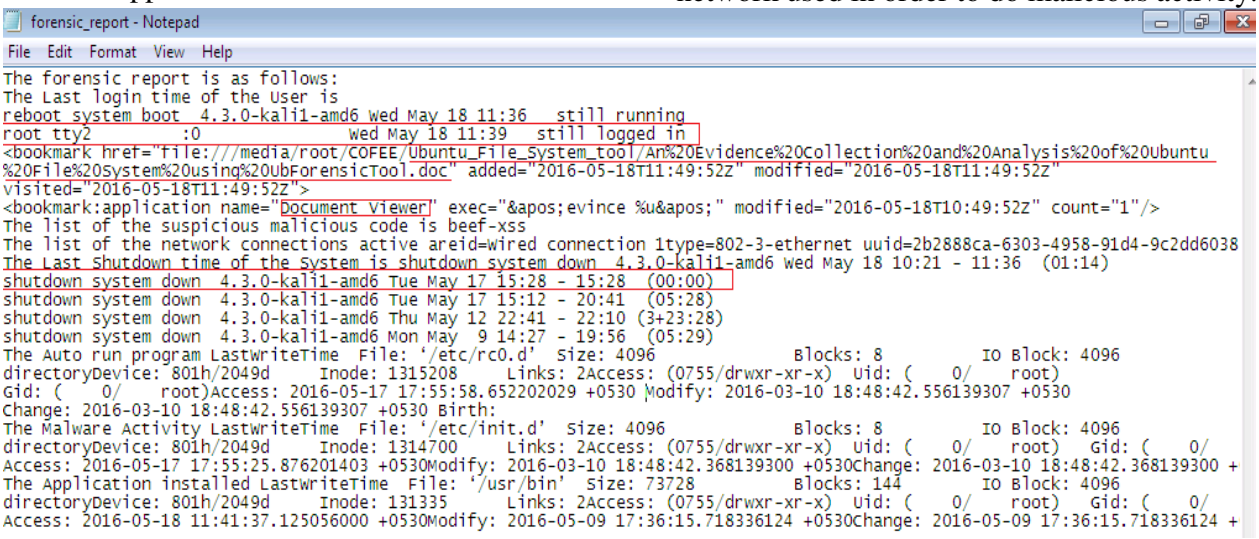

Figure 3. Forensic report using UbuntuForensic tool

Devices connected to the System

In Ubuntu "lshw" command provides the list of hardware devices attached to the system. Also, the /dev directory in the file system provides the information about the hardware attached to the system. The syslog file also maintains the details of the devices which have been detected.

The date and timing at which the device was connected along with device details are also recorded in the syslog.

Last Login Activity of the user

In Ubuntu, the login time and the logout time can be accessed by using the 'last' command at the terminal. Syslog file in the /var/log maintains the login and shutdown time.

Malware Activity

To remain running after reboots, malware is usually re-launched using some persistence mechanism available in the various startup methods on an Ubuntu system, including services, drivers, scheduled tasks, and other startup locations. There are several configurations files that Ubuntu uses to automatically launch an executable when a user logs into the system that may contain traces of malware programs. Malware often embeds itself as a new, unauthorized service. A certain amount of malware use /etc/init.d directory to hide and start their execution on startup of the system.

3.3 Timeline Analysis

The digital forensic investigator should detect the activity being performed by the suspect along a timeline. By performing the timeline analysis, the investigator can trace the sequence of events that were performed by the suspect. For instance, if the suspect had accessed a word document by logging using a login id, the date and time of these activities can be correlated to convict the suspect. The forensic report obtained as in Figure 3 shows root user had logged in at 11:39AM on 18/05/2016 and accessed the .doc file 'An Evidence Collection and Analysis of Ubuntu File System using UbForensicTool' at 11:49AM using document viewer application. This forensic information can be evidence against the root user for accessing the .doc file as the .doc file was accessed after the login time by root user and before the shutdown of the system. The forensic report thus obtained using the UbuntuForensic tool underlines the importance of performing the timeline analysis of the activities.

3.4 Data Security

The approach for creating the backup of the data and identifying the modified data on the hard disk is proposed as follows.

The UbuntuForensic tool provides the facility for the backup of the files from the hard disk of the running system. The backup of these files is maintained on the external storage media. The content of these files then can be hashed one by one and the resulting hashes are then indexed and stored along with the file name and the path of the file on the hard disk in a table on the external storage. The md5 algorithm can be used to obtain the hashes from the backup data.

In order to detect if any changes to the data on the hard disk of the running system have been caused by the suspicious criminal, the hashes are obtained from the individual files on the hard disk one by one and these hashes are then compared with the hashes stored on the external storage media. The comparison of two hashes is performed only if the entry for a particular file name on the hard disk is found in the table on the external storage. Otherwise, the concerned file is considered as deleted by the suspicious user and a report can be prepared regarding this. If two hashes which are being compared are found dissimilar then it means that the criminal has caused some modification to the relevant file on the hard disk. A report can be prepared about all the files whose hashes are found dissimilar from that of the hashes in the external storage. In such situation, the affected file can be restored back from the external hard disk.

The Figure 4 depicts the process for detecting the modification of the data on the hard disk by the criminal.

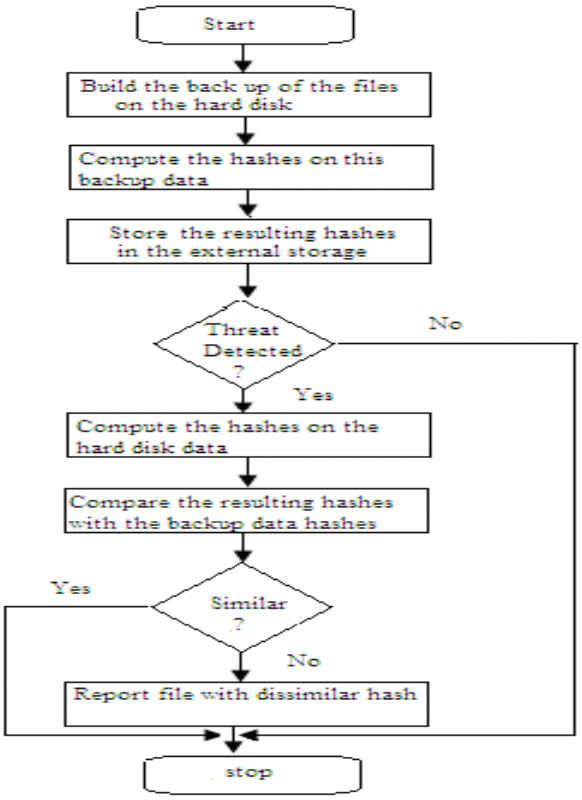

Figure 4. Flowchart depicting operation for identification of modified files using UbuntuForensic tool

4 SOFTWARE ARCHITECTURE AND IMPLEMENTATION

The software architecture of the UbuntuForensic tool is illustrated in Figure 5.

The analysis of local and the external hard disk directory structure can be performed using the UbuntuForensic tool. The evidence and time of the activity are extracted and the report is generated for correlating the sequence of events and their timings.

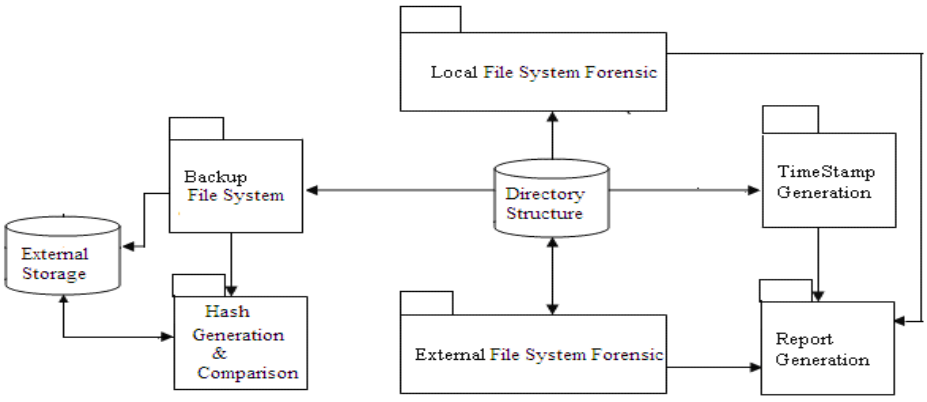

Figure 5. Software Architecture of UbuntuForensic tool

The software architecture consists of following modules: Local File System Forensic, External File System Forensic, Timestamp Generation, Backup File System, Hash Generation and Comparison, and Report Generation.

The Local and External File System Forensic deals with extracting forensic evidence for various user activities from the directory structure of the system on which the tool is running and the directory structure available on the external hard disk. The time stamp generation module generates the last modified timestamp for the directory and the files associated with the user's activity concerned. The forensic Report based on the forensic evidence obtained and the generated timestamp is obtained using Timestamp Generation module.

The File System forensic algorithm for the proposed tool is as follows:

Requires Activity (i, D(DIR)) returns the extracted forensic information forensic_info for each i^{th} activity from the DIR directory of the directory structure D. Select(forensic_info(i)) selects the evidence from the forensic_info. Timestamp (i, D(DIR)) returns the timestamp for the directory DIR for the i^{th} activity. Generate_Report generates the report from the selected evidence and the timestamp. MAX indicates the maximum number of user's activity.

Input The directory structure D

Output Report in text format

1. For i ϵ (1, MAX) do;
2. forensic_info(i) ← Activity(i,D(DIR))
3. forensic_evidence(i)←select(forensic_info(i))
4. $timestamp_i$ ← Timestamp(i,D(DIR))
5. Report←Generate_Report(forensic_evidence, timestamp)

The Activity(i,D(DIR)) function extracts the forensic information from the directory structure for the i^{th} activity of the user. Once the forensic information is extracted, the forensic investigator can select the digital evidence from it. The Timestamp(i, D(DIR)) function generates the timestamp for the i^{th} activity of the user based on the last access and modification timestamp of the directory. As the contents of the directory are accessed or changed, the timestamp of the directory also gets changed. This procedure is repeated for all the users' activity in consideration. Once all the activities are finished, the forensic investigator generates the Forensic report.

The backup of the files managed by the file system is performed using Backup File System module. The data backed up is then hashed by the hash generation module to generate the md5 hash. The hash so obtained then can be stored on the external storage in a relational table. Whenever the threat is detected, the hashes can be obtained for the hard disk data and these hashes then can be compared with the hashes in the external storage to identify the modified files by the criminal as discussed in section 3.4. The structure definition of the table storing the hashes on the external storage is proposed as follows:

```
typedef struct  _TABLE
{
Number      int;
File_Name  string[20];
Path_Name string[20];
Hash long int;
                } table;
```

The field description is as follows:

- Number: This field is an index for the entry in the relation.
- File_Name: The name of the backed up file from the hard disk.
- Path_Name: The path of the concerned file on the hard disk.
- Hash: The md5 hashes obtained on the content of the file.

The UbuntuForensic tool is built using QT4, a cross-platform application frame-work that is widely used for developing application software that can run on various software and hardware platforms with little or no change in the underlying code base while having the power and speed of native applications. Qt uses standard C++ with extensions including signals and slots that simplify handling of events, and this helps in the development of both GUI and server applications which receive their own set

of event information and should process them accordingly. The UbuntuForensic tool uses QSetting class and its methods to extract the information's from the directory structure of the Ubuntu file system.

5 EVALUATION

The comparison between the existing widely used Linux forensic tools and the UbuntuForensic Tool is performed as in table 1. The tool like TSK, autopsy can list file and directories and perform timeline analysis of file activity. DEFT and CAINE provides GUI based forensic tools. i-Nex and History tools provide information about the hardware connected to the system and the recent command executed on the system recently, respectively. However, it has been observed that none of the Linux tools provides the facility for extracting the evidence for the specific activity of the user. Comparatively, the UbuntuForensic tool performs the extraction of forensic related information about the various users' activity being performed on the system. The UbuntuForensic tool also performs timeline analysis using which the conviction of the criminal can be performed based on the last access, modification dates of the directories and the login time of the suspicious user. The UbuntuForensic tool supports local and external file system forensics. In External file system forensics, the external hard disk with Ubuntu operating system is mounted on the system with the UbuntuForensic tool to extract the forensic evidence.

The proposed UbuntuForensic tool also performs the backup of the files and directories and also provides hashing of the file contents to identify any changes to the file by the criminal. Based on the advanced requirements mentioned in the paper, UbuntuForensic tool improves over the shortcoming of the existing tools.

The UbuntuForensic tool is tested on 5 hard disks with Ubuntu and Linux compatible file system. These disks are classified into two sets: internal and external. The internal hard disk is an indispensable part of the Computer System on which the UbuntuForensic tool is running. The external hard disk is needed to be connected externally to the system to extract the evidence from it. The disk1, disk2, disk4 are internal hard disk and the disk3 and disk5 are external hard disk. The disk1 partitions are formatted with ext2 file system. Disk2 and disk4 partitions are formatted with ext3 file system. Disk3 and disk 5 partitions are formatted with ext4 file system. The effectiveness of the tool is obtained in terms of Retrieval Rate metric for extracting the evidence for the various users' activities. It has been observed that the effectiveness of the tool is 100% for all the hard disks used in the experimentation for Autorun, Recently Accessed Documents, applications installed, Last Login, Malware activities. However, in the case of Network Connected and Devices Connected activities, the Retrieval Rate is 98% as the command such as netstat and lshw displays the network and hardware devices related information only about the running system.

Table 1. Functional comparison with existing tools

Tool	**Function**				
	Integrated Analysis	Timeline Analysis	Activity Analysis	GUI support	Any other feature
UbuntuForensicTool (Proposed)	✓	✓	✓	✓	Running process, Hash Generation
The Sleuth kit(TSK)	X	✓	X	X	Recovers deleted files
Autopsy	X	✓	X	✓	Recovers deleted files
Scalpel	X	✓	X	X	Recover data from disks
DEFT	X	✓	✓	✓	Data Recovery and hashing, Process information
CAINE	X	✓	✓	✓	Data Recovery
i-Nex	X	✓	✓	✓	Display device information, generate report
History	X	X	✓	X	Lists only command history

Hence these commands can't extract this information from the external hard disk (disk3). The effectiveness of the tool is summarized in figure 6.

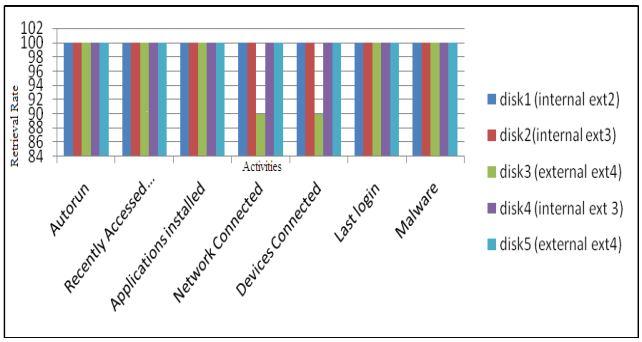

Figure 6. Effectiveness of UbuntuForensic tool

6 CONCLUSION

The File System maintains historical information about user activity in its directory structure. All of this information can be extremely valuable to a forensic analyst, particularly when attempting to establish the timeline of activity on a system. It is essential to perform the analysis of the file system and use timeline analysis to detect the suspicious activities of the suspect. A wide range of cases would benefit greatly from the information derived or extracted from the file system.

A survey on the existing Linux forensic tools revealed that they extract very little forensic information from the file system. Comparatively, the UbuntuForensic tool provides more evidence from the file system as that of the existing tools; saving the time and effort in searching the evidence. The UbuntuForensic tool also covers forensic analysis of the file system on the external hard

disk, thus enabling the forensic investigator to conduct the forensic investigation without changing the setup. By computing the hashes on the files from the hard disk, it is observed that the files which are modified by the criminal can be identified.

7 REFERENCES

1. SophosLab, "Botnets, a free tool and 6 years of Linux/Rst-B", https://nakedsecurity.sophos.com/2008/02/13/botnets-a-free-tool-and-6-years-of-linuxrst-b, 2008.
2. Sophos, "Don't believe these four myths about Linux Security", http://blogs.sophos.com/2015/03/26/dont-believe-these-four-myths-about-linux-security, 2015.
3. J. McInnes, "Linux Operating System don't get attacked by viruses,why?", https://www.quora.com/Linux-Operating-System-dont-get-attacked-by-Viruses-why, 2015.
4. L. Tang, "The study of Computer forensics on Linux", International conference on computational and Information Sciences , 2013.
5. Y. Kuo-pao and K. Wallace, "File Systems in Linux and Free BSD:A Comparative study", Journal of Emerging Trends in Computing and Information Sciences,vol.2, 2011.
6. C.Wei and L. Chun-mei, "The Analysis and Design of Linux File System Based on Computer Forensic", International Conference on Computer Design and Applications , 2010.
7. C. Joonah, C. Antonio, G. Paolo, L. Seokhee and L. Sangjin, "Live Forensic Analysis of a Compromised Linux System using LECT(Linux Evidence Collection Tool)", International Conference on Information Security and Assurance, 2008.
8. B. Grundy, "Advanced artifact analysis", European Union Agency for Network and Information Security, 2014.
9. "ArchLinux", https://wiki.archlinux.org/index.php/List_of_application/Utilities, 2016.
10. D. Patil and B. Meshram, "Forensic investigation of user activities on Windows7 and Ubuntu12 operating system", International Journal of Innovations in Engineering and Technology, vol. 5, 2015.