Exploring Digital Forensics Tools in Backtrack 5.0 r3

Ahmad ghafarian[1] and Syed Amin Hosseini Seno[2]
[1] Department of Computer Science, University of North Georgia, Dahlonega, GA USA
[2] Computer Networks Laboratory, Department of Computer Engineering, Ferdowsi University of Mashhad, Iran

Abstract - *Computer forensics tools are essential part of any computer forensics investigation. The tools can be classified in various ways including, open source vs. proprietary; hardware vs. software; special purpose vs. general purpose, etc. In practice, software tools are more common. Each software tool has its own pros and cons. However, they all have one feature in common, i.e. installation, configuration, and setup. For some tools, the configuration process can be complicated and time consuming. To avoid this, the computer forensics investigators have the option of using the computer forensics tools that are pre installed and configured in Backtrack 5.0 r3. In this paper, we present the results of our experiment with various digital forensics tools that are included in Backtrack 5.0 r3.*

Keywords: Backtrack, VMware, Computer Forensics Tools

1 Introduction

Computer forensics tools play an important role for forensics investigators. Selection of a particular tool depends on the nature of the investigation, reliability, security, and the cost effectiveness. There are many options that digital forensics investigators can choose from. Classifications of computer forensics tools include open source, proprietary, hardware, software, special purpose and general purpose. Each tool has its own advantages and disadvantages. A comprehensive review of the top twenty open source free computer forensics investigation tools can be found in [14]. For a list of proprietary computer forensics tools see [16] & [9]. Brian Career [3] reports on how forensics tools have been viewed historically, i.e. philosophy, security and reliability. He concludes that open source tools are as effective and reliable as proprietary tools. Manson and his team [8] compared one open source tool and two commercial tools. They found that all three tools produced the same results with different degree of difficulty. Backtrack 5.0 r3 has a rich repository of digital forensics tools that support computer forensics specialists to do tasks such as acquisition, analysis, recovery, imaging, vulnerabilities scan, penetration testing, and file interrogation. A survey of Backtrack 5.0 network forensics tools can be found in [7]. The purpose of this research is to study Backtrack 5.0 r3 [2] forensics tools. We examine different categories of computer forensics tools, analyze the types and number of tools in each category, investigate their capabilities, evaluate their effectiveness, and present the result of our experiment with all the available tools in Backtrack 5 r3. In the next section we discuss the platform for our e.

2 Our Virtual Machine Platform

Backtrack is a Linux based operating systems that comes with a rich repository of security and forensics tools [2]. The computer forensics tools are grouped into several categories. We use the forensics tools within the Backtrack.

VMware Workstation is a hypervisor that runs on 64-bit computers [15]. It enables us to set up multiple virtual machines and network them together. Each virtual machine can execute on different distribution of Linux operating system. VMware Workstation is proprietary software but we used the trail version for free. Below are the steps for setting up the platform for our experiment.

1. Install VMware Workstation on a machine
2. Create a virtual machine on the VMware workstation
3. Install Backtrack 5.0 r3 on the virtual machine
4. Launch Backtrack 5.0 r3 from the virtual machine
5. From the list, select forensics and then select a tool

3 Forensics Tools Experiment

There are several categories of computer forensics tools in Backtrack. Some categories have more than one tool. In the following subsections, we explore the details of the tools. For each tool, we review its purpose, the syntax for running the tool and the results of executing the tool on our virtual machine platform.

3.1 Anti Virus Forensics

The tools in this category include *Chkrootkit, and rkhunter*

3.1.1 Chkrootkit

Chkrootkit is a program that checks for signs of rootkit infection on a machine during live acquisition. It runs on almost all versions of the Linux. Depending on the option selected by the user, *Chkrootkit* can perform an individual infection scan, *sshd* infection test, as well as full scan. Network administrators can also use it to check for known rootkits. We ran this tool on our virtual machine by issuing the

following command. */pentest/forensics/chkrootkit –x/-q* where switches, - x and - q indicate expert mode and quiet mode respectively. It only took a couple of minutes to execute and present the report. It reported no rootkit in our virtual machine as expected.

3.1.2 Rkhunter

Rkhunter is another utility which can be used in live acquisition to check for signs of rootkits on Linux based systems. It is a rich scanning tool that scans for rootkits, backdoors, local exploits, hidden files and comparing MD5 hashes. We executed this tool on our virtual machine for a full scan using the command: */pentest/forensics/rkhunter -c –sk*. It examined 163 files and applications and 8 suspects file were identified.

3.2 Digital Anti Forensics

TrueCrypt is the only tool in this category. It is able to establish and maintain an on-the-fly-encrypted volume. This means that data is automatically encrypted right before it is saved and decrypted right after it is loaded, without any user intervention. To read data from an encrypted volume we must use an encryption key. This tool encrypts the entire file system e.g., file names, folder names, contents of every file, free space, metadata, etc. TrueCrypt currently supports the following hash algorithms: RIPEMD-160, SHA-512 and Whirlpool. This utility is not pre installed on Backtrack 5.0 In our experiment; we installed the TrueCrypt, mounted the volume on VMware virtual machine, and then executed TrueCrypt. To see the effect of TrueCrypt, we saved some Microsoft office files on the mounted volume. When we tried to retrieve the files TrueCrypt asked for the encryption key. Upon entering the encryption key, the files were opened successfully.

3.3 Digital Forensics

Hexedit is a digital forensics tool which has the capability to view and edit files in hexadecimal or in ASCII format. Some features of *Hexedit* include, reading a device as a file, comparing two files, searching, and statistical calculations on the data of a file. With Hexedit being activated, we used the command *media/ashrafian/test.dd* to examine the content of the *test.dd* file which is saved in *Ashrafian* folder. The result is shown in Figure 1 below.

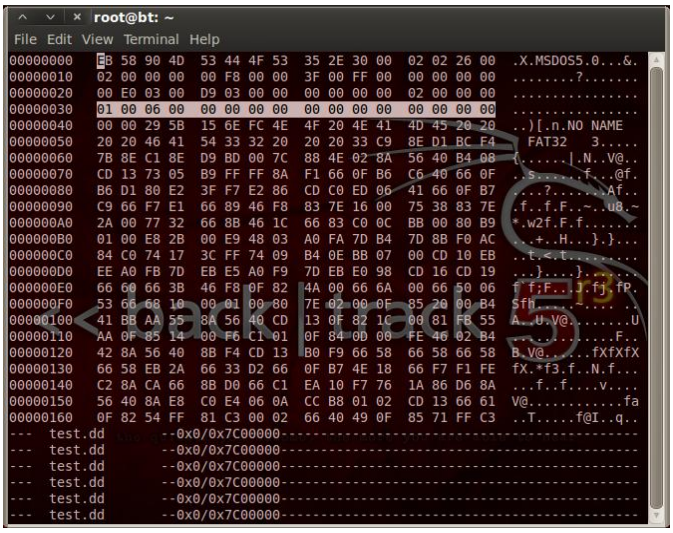

Figure 1- Statistical analysis of test.dd file using Hexedit

3.4 Forensics Analysis Tools

The tools in this category include *bulk_extractor*, *evtparse.pl*, *exiftool*, *misidentify* and *stegdetec*

3.4.1 Bulk_explorer

Bulk_extractor is a program that can scan a disk image to search for personal data such as credit card numbers, email addresses, domain names, urls, telephone numbers, text messages, etc. It can also automatically detect, decompress, and recursively re-process compressed data. It is capable of processing data from various devices such as hard drives, optical media, camera cards, cell phones, network packet dumps, etc. It classifies outputs information in various files such as *ccn.txt* for credit card numbers, *domain.txt* for Internet domains found in the file, *email.txt* for email addresses, *exif.txt* for exif data from media files and *wordlist.txt* for the list of all words extracted from the file.

We used *bulk_explorer* to scan a USB flash drive. The command to execute it is: *bulk_extractor - o outputdir /media/A.dd* (where *outputdir* is a directory and *A.dd* is a file that contains image of the drive under investigation.) We stored some personal information in the USB to demonstrate the *bulk_extractor*'s behavior. It turned out that the data that was stored in the USB was retrieved in the corresponding output files. Upon examination of these files, we were able to see the original data that we saved in the USB. The extraction of data was very fast. This speed is attributed to the fact that *bulk_extractor* can scans different parts of a file in parallel and thus no need for file parsing or any knowledge of the file system. See Figure 2 below.

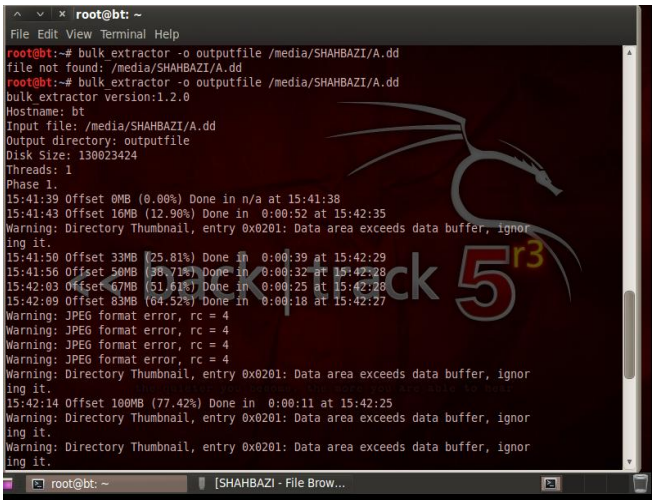

Figure 2- Execution of bulk_explorer output

3.4.2 Evtparse.pl

Evtparse.pl is a Windows event file parser utility. It generates a text output from the event files which may contain useful information. It is a useful tool for work with event files such as Windows log file. We applied this utility on a Windows log file called *A.evtx*. It produced information such as date file was accessed, time, etc. The format we used is: */ evtparse.pl –e /media/Shahbazi/A.evtx*, where Shahbazi is a folder that contains A.evtx file.

3.4.3 Exiftool

Exiftool is a command line utility that allows users to read or write metadata to image files. To retrieve metdata from A.dd image**,** we used the command: *Exiftool - a /media/Shahbazi /A.dd.* As can be seen from Figure 3; Exiftool extracted metadata from the image file. As we can see from the figure, important metadata infomration are listed.

Figure 3-The result of running Exiftool on A.dd file

3.4.4 Misidentify

This utility can be used to find Windows 32 executable files recursively. We launched this utility to list executable files in a USB drive by using the command *misidentify -r /media/Shahbazi/forensic*. Since our virtual machine is Windows 64, there was no Windows 32 executable file in the forensics folder.

3.4.5 Stegdetect

This tool will look for signatures of several well-known steganography embedding programs in order to alert the user that text may be embedded in the image file, such as jpeg. To see if there is steganography embedded message in our AA.jpg file in a USB drive, we launched *Stegdetect* by using this command *stegdetect –t /media/Shabazi/AA.jpg*
The result of executing the utility is shown below

stegdetect –t /media/Shabazi/AA.jpg: negative

Where *negative* indicates no message embedded in the AA.jpg file.

3.5 Forensics carving tools

The tools in this category include *Foremost, recoverjpeg, safecopy, scrounge-ntfs, & Testdisk*

3.5.1 Foremost

Foremost is a popular file carving utility. It takes image files to search for file headers in order to recover files. The carving process utilizes attributes such as unique signatures, file headers, and file footers. Some limitations of *foremost* is when files are fragmented, header is overwritten, it is a common string, or is changed due to actions such as compression. *Foremost* configuration file also allows the forensic examiner to customize the types of files that will be recovered and enables the use of wildcards for pattern matching. *Foremost* opens image file in read-only mode, which is important for maintaining the forensic integrity. It can handle both Windows and Linux file systems. We applied *Foremost* to search the image *Foremost.dd* for jpeg files. The command we used for this process is:

Foremost jpeg -o/ root/Desktop/media/Shahbazi/foremost.dd

Where switch - o specifies the output directory for storing recovered files. As can be seen from Figure 4, the recovered jpeg files are listed. The file size and dates are also displayed.

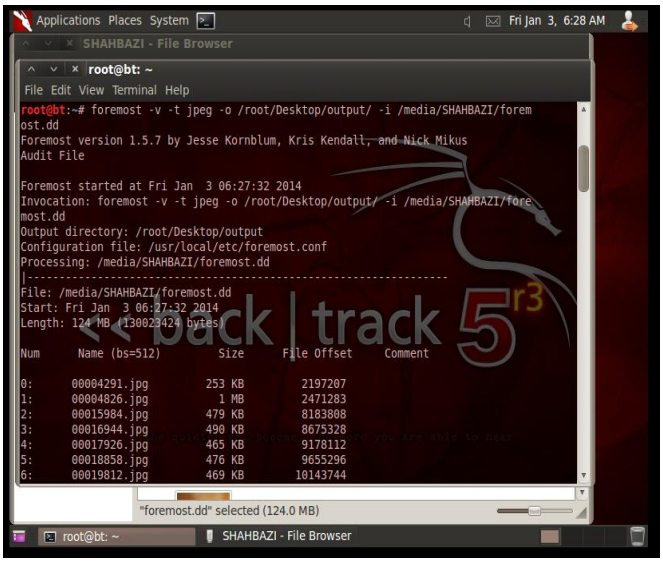

Figure 4-*Foremost* recovered jpeg files

3.5.2 Recoverjpeg

This is another utility for recovering jpeg images from a file system. We also used this tool to recoverjpeg images from foremost.dd folder on USB drive by issuing this command: *recoverjpeg/media/Shahbazi/foremost.dd*. The same Jpeg files that are shown in Figure 5 were retrieved.

3.5.3 Safecopy

This utility can recover as much data as possible from a damaged device, such as a hard drive or a USB drive. Other programs such as dd, cat, or cp will stop reading data once a damaged area is hit, while Safecopy will read to a point designated by the user, regardless of damaged areas. It does this by identifying the damaged areas, and skipping around them. To recover data from a damaged USB drive, we used: *Safecopy/media/Shahbazi/root/Desktop/rescue files*
With this tool, we were able to recover files to rescue folder that exist on /root/Desktop.

3.5.4 Testdisk

Testdisk is a program that specializes in recovering lost disk partitions and making disks bootable. It has the ability to rebuild partition tables, rebuild boot sectors, fix the Master File Table, and recover deleted partition and files. Our experiment with Testdisk reported back no error in our system.

3.6 Forensics Hashing Tools

Backtrack 5.0 supports many hashing utility including *Hashdeep, MD5deep, Sha1deep, Sha256deep, Tigerdeep,* and *Wirlpooldeep*

All the tools listed above basically do the similar job, i.e. calculating the message digest of an input file. Each utility is a suite of cross platform tools to compute and compare MD5, SHA-1, SHA-256, Tiger, or Whirlpool message digests on an arbitrary number of files. We used *MD5deep* to calculate the hashes of all files in our input folder. The command to do that is: *md5deep /root/Desktop/output/* where the message digests is saved to a file in a directory called *output* (see Figure 5).

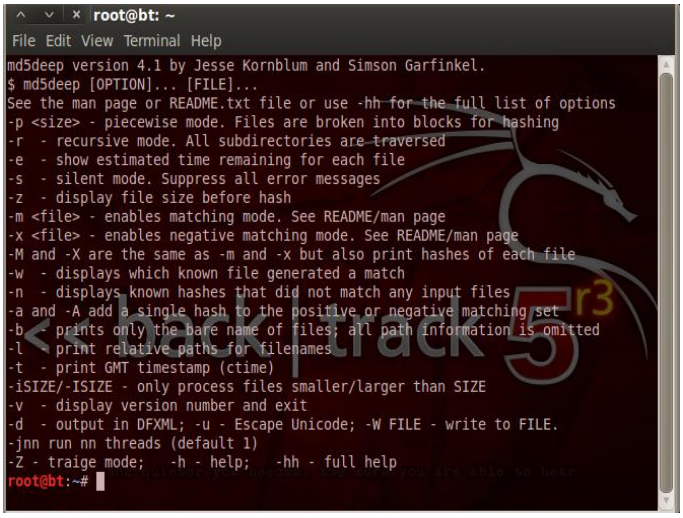

Figure 5-MD5deep running on Backtrack

3.7 Forensics Imaging Tools

There are several imaging tools in Backtrack 5.0. These include *Air, Dc3dd, Ddrescue, & Ewfacquir**e.*** In the next subsections we report their performance.

3.7.1 AIR

AIR (Automated Image and Restore) is a utility which can be used to create forensics bit images from device drives. AIR supports MD5/SHAx hashes, SCSI tape drives, imaging over a TCP/IP network, splitting images, and detailed session logging. AIR itself is a GUI interface for dd/dc3dd. On Backtrack 5.0, when we first selected AIR, it downloaded and compiled the necessary components for running the program. Then we followed these steps to create an image of a device:

- Choose USB1 as the source device
- Choose USB2 as the destination device.
- Choose Block size of the source and the destination 512 Byte.
- Choose MD5 as the hash method.

After these steps, when we clicked at the start the imaging begins (see figure 6.) It took several minutes to create an image USB1 in USB2.

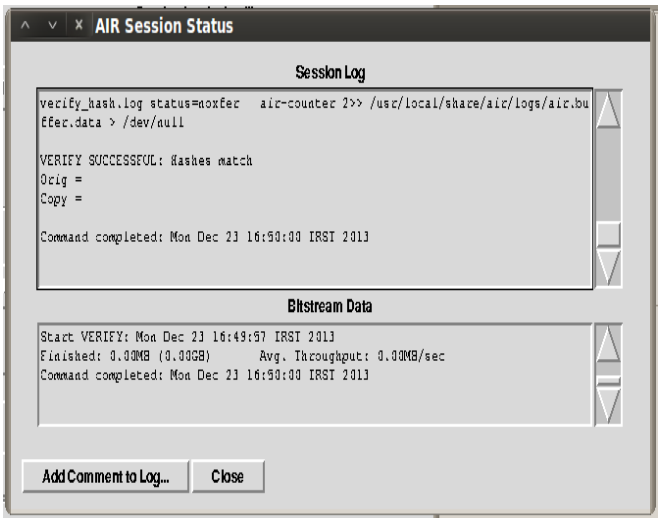

Figure 6-AIR Session Status

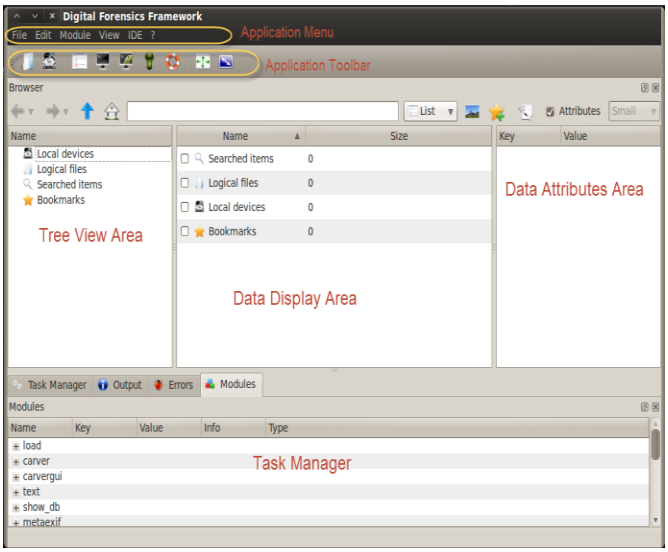

Figure 7 - DFF interface

3.7.2. D3dd

D3dd is an altered version of dd, that we used to operate low level disk functions. We used *D3dd* to split a large disk image into smaller pieces. For the input file of */dev/sda* it calculated hashes for the individual new files and the original large file was broken into 2 GB pieces with "000″ as a suffix in the filename. It also saved logs of all data to */root/Desktop/log.txt*, and output the smaller files to */root/Desktop/images*.

3.7.3 Ewfacquire

Ewfacquire can be used to create disk images in the EWF format. It includes several message digests including MD5 and SHA1. To create an image of */dev/sdb1* and logging data to */root/Desktop/log.txt*, we obtained the image by issuing this command on Backtrack

ewfacquire -d sha1 -l /root/Desktop/log.txt /dev/sdb1

3.8 Forensics Suites

We applied *DFF (Digital Forensics Framework)* to collect, preserve, and reveal digital evidence. In Backtrack 5.0 we can launch it via Forensics Suites menu. In order to run *DFF*, we first loaded an evidence file, i.e. a forensic image that we created using one of the previous tools. *DFF* then processed the evidence file against one of the built-in modules to begin analyzing data. Figure 7 below shows the *DFF* analysis of the evidence.

3.9 Network Forensics

Tools in this category include *Driftnet, P0f, Tcpreplay, Xplico, & Wireshark*

3.9.1 Driftnet

Driftnet is a network utility that sniffs traffic for images and other media. Rather than sniffing all traffic using utilities like Wireshark, *Driftnet* makes it easier by automatically picking out images and media. We used the command *driftnet -i eth0 -v*, to capture traffic and instruct *driftnet* to be verbose mode in its output. The result produced useful information which is valuable to forensics investigators.

3.9.2 P0f

p0f is a passive operating systems fingerprinting tool. All the host has to do is connect to the same network or be contacted by another host on the network. The packets generated through these transactions gives *p0f* enough data to guess the system. In our experiment, by issuing the command *p0f -f /etc/p0f -i eth0*, we were able to read fingerprints from /etc/p0f and listen on eth0 via *libpcap* application.

3.9.3 Xplico

Xplico is a Network Forensic Analysis Tool (NFAT) that is capable of extracting application data from packet capture files. It is best suited for offline analysis of PCAP files but it can also analyze live traffic. *Xplico* can extract email, HTTP, VoIP, FTP, and other data directly from the PCAP files. It is able to recognize the protocols with a technique named Port Independent Protocol Identification (PIPI). We executed *Xplico* in Backtrack 5.0 by issuing the following commands:

Start Xplico - /etc./init.d/xplico
Go to http://localhost:9876
login with default user and password
user name: xplico
password: xplico
Click the new case

In *Xplico* a case is composed of one or more sessions. Figure 8 shows a new case we have created in *Xplico*. This session captured traffic for offline analysis of PCAP files. The captured traffic was analyzed and no unusual activities were found.

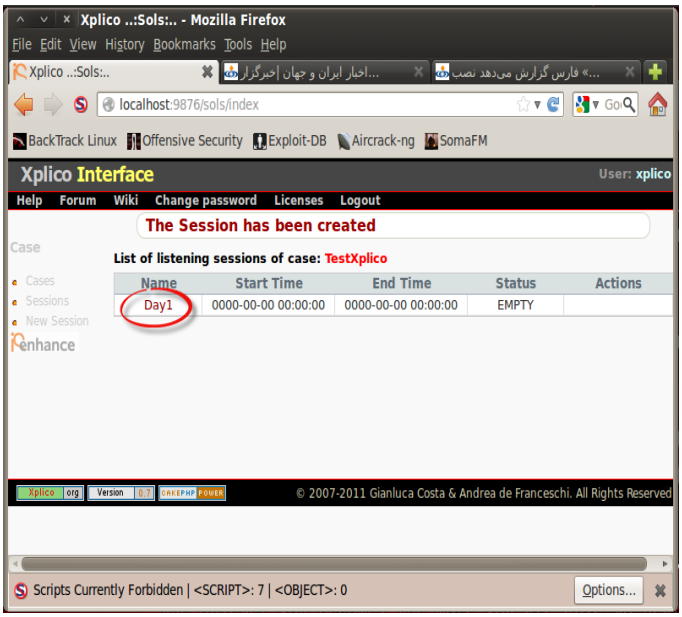

Figure 8 - Session Creation in Xplico

3.10 Password Forensics Tools

The tools in this category include *CmosPwd*, *fCrackZip*, and *Samdump*. All these tools are password cracking and or retrieving tools. In the next subsection we describe the result of the application of *FCrackZip*.

3.10.1 FCrackZip

FCrackZip is a tool for breaking the password of a password protected zip file with a brute force or dictionary attack [6]. When using this tool in brute force mode we can specify the length, character types, and initial strings for the password. Before we launch this tool we need to upload a password protected zip file to the Backtrack. By default, brute force starts at the given starting password, and successively tries all combinations until they are exhausted. Then, it prints all passwords that it detected. We applied brute force attack for cracking the password protected *srl.zip* file (see Figure 9):

FCrackZip -b -l 2 -c 'aA1' /root/Desktop/srl.zip
Where

-b > brute force
-c aA1 > char set lower, upper, alphabet
-l > length of expected password

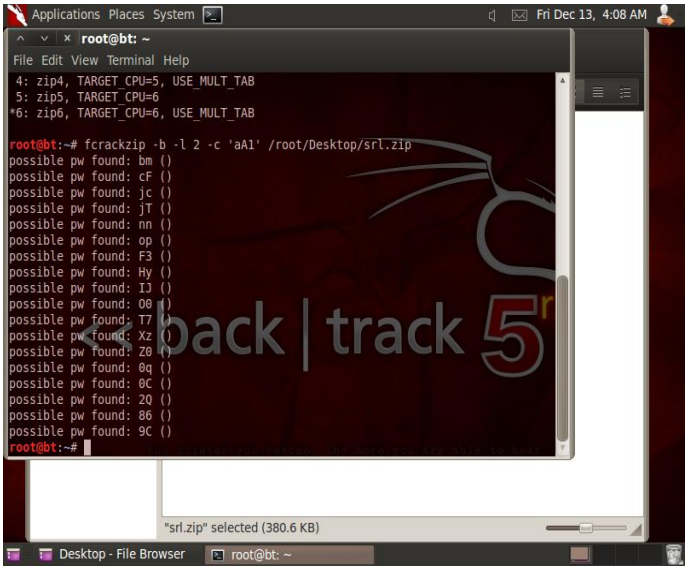

Figure 9- FCrackZip cracking a password protected zip file

We should note, depending on the password combination, this tool may take a long time to process even for a small file.

3.11 PDF Forensics Tools

The utilities in this category include *PDFid, PDF Parser, and Peepdf*

3.11.1 PDFid

Pdfid is a utility that can extract useful information from a PDF file. *Pdfid* can show forensics investigators any suspicious activities in the PDF files. It can also scan a PDF file to look for certain PDF keywords such as JavaScript. The execution of *pdfid.py* on file.pdf produced useful information about the file such as header, date, etc.

3.11.2 PDF Parser

PDF Parser is an investigation tool that can be used to examine some of the content within a PDF file. When some characters are discovered that appear to have no meaning, then other tools such as ASCIIhexDecode can be used. We used the command *pdf-parser.py -a /mnt/shared/nw.pdf*
to display statistics about the new.pdf file (See Figure 10.)

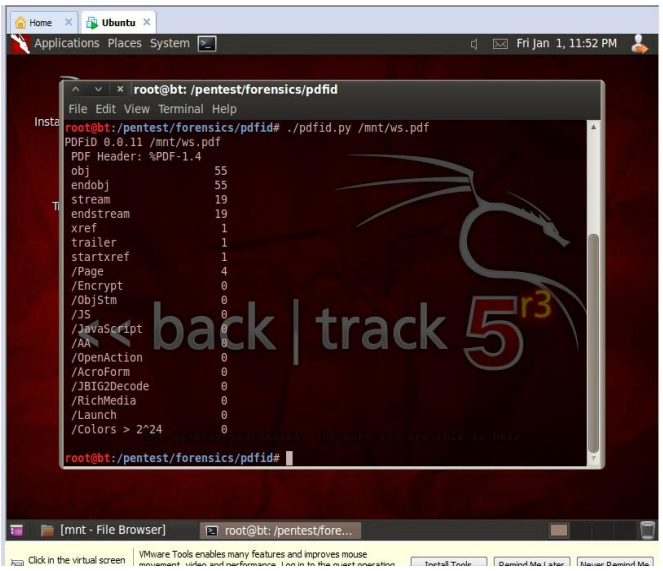

Figure 10 - Result of running PDFid on a PDF file

4 Conclusions

In this work we have demonstrated the application of various computer forensics tools on Backtrack 5. We showed the syntax for using the tools including applicable switches, and the result of executing the tools on our virtual machine. As it was demonstrated the tools produce consistent results according to their specifications. However, similar results can be obtained by using physical machines. Our results will help the computer forensics investigators on selecting appropriate tool for a specific purpose. It also helps penetration testers to check for signs of vulnerabilities on their system. We showed that Backtrack 5 is a good choice for forensics investigators for several reasons. These include, the tools are free, easy to use, no need for configuration, and produce consistent results.

5 Future Research

To extend this research, we intend to gather more detailed instructions of the tools for potential users of the tools. In addition, we plan to install Backtrack on a physical machine and perform the same experiments. We also plan to use selected non-Backtrack open source computer forensics tools, observe their performance and compare the results with the same tools in Backtrack 5.0. Another area which is worth further study is RAM forensics for social networking sites such as Facebook.

6 References

[1] Alex, (Jan 2013), *exiftool-Backtrack 5-Forensics-Digital Forensics Analysis-exiftool*, Question Defense, http://www.question-defense.com/2013/01/02/exiftool-backtrack-5-forensics-digital-forensics-analysis-exiftool

[2] Backtrack 5 r3, http://www.backtrack-linux.org/backtrack/backtrack-5-r3-released/

[3] Carrier, B (Oct 2002), Open Source Digital Forensics Tools: The legal argument. http://dl.packetstormsecurity.net/papers/IDS/atstake_open source_forensics.pdf

[4] De Smet, D, & Willie L. Pritchett (March 2013), *Backtrack Forensics*, Cookbooks Networking & Telephony Open Source, http://www.packtpub.com/article/backtrack-forensics

[5] Gupta, A, *Digital Forensics Analysis using Backtrack, Part 1 & 2* http://www.linuxforu.com/2011/03/digital-forensic-analysis-using-backtrack-part-1/

[6] Lazzez, Amore (January 2013), A survey About Network Forensics Tool, International Journal of Computer and Information Technology, Vol 2, Issue 1, pages 74-81.

[7] List of Tools in Backtrack http://secpedia.net/wiki/List_of_tools_in_BackTrack

[8] Manson, D; Carlin, A. ; Ramos, S. ; Gyger, A. ; Kaufman, M. ; and Treichelt, J (2007*). Is the Open Way a Better Way? Digital Forensics Using Open Source Tools, System Sciences*. HICSS 2007. 40th Annual Hawaii International Conference on Science, Page 266b-270.

[9] Mares and Company (2013), Alphabetical list of links to manufacturers, suppliers, and products, http://www.dmares.com/maresware/linksto_forensic_tools.htm

[10] Nelson, B; Phillips, A; Stuart, C., (2010), *Guide to Computer Forensics and Investigation*, 4ed, Cengage Learning.

[11] Nolan, R, Colin O'Sullivan, Jake Branson, Cal Waits, *(March 2005), First Responders Guide to Computer Forensics*

[12] Rose, M (July 2006) *Brute force cracking*, http://searchsecurity.techtarget.com/definition/brute-force-cracking

[13] Sigh, G, Crack the password protected zip files using fcrackzip-Backtrack, http://hackthedark.blogspot.com/2012/06/crack-password-protected-zip-files.html

[14] Tabona, A. Z. (2002), *Top 20 Free Digital Forensics Investigation Tools for SysAdmins*, http://www.gfi.com/blog/top-20-free-digital-forensic-investigation-tools-for-sysadmins/

[15] VMware Virtualization for Desktop & Server, Application, http://www.vmware.com/training/

[16] Wikipedia, List of Digital Forensics Tools, http://en.wikipedia.org/wiki/List_of_digital_forensics_tools