Research on UNIX Forensic Analysis

Lianfu Yin
Nahu College, Jiaxing University
Jiaxing, Zhejiang 314033, China
ylf@mail.zjxu.edu.cn

Abstract—**UNIX is one of the most mainstream operating systems, it has great practical significance to research the methodology of UNIX forensic analysis. This paper firstly introduces the method to capture the volatile data from UNIX systems, then introduces the concrete steps and method of UNIX forensic analysis.**

Keywords- computer crime; computer forensics; UNIX forensics Analysis

Ⅰ. INTRODUCTION

As UNIX is one of the most common mainstream operating systems, it is quite important to analysis the forensic method of UNIX system. Before we do forensic analysis, we shall obtain forensic data first. Forensics data are divided into two main categories, one is volatile data, and the other is nonvolatile data.

Volatile data are those data which will all disappear when the computer shuts down. These data are usually in internal memory, mainly including information like the status of the network connection, the state of a running process, and so on. The so-called initial response refers to the collection of volatile data on the victim machine, and the forensic analysis process. Non-volatile data are those data still exists when the computer shuts down, and these data are generally on the hard disk.

This paper firstly introduces a method for obtaining volatile data from an UNIX system, and then introduces obtaining hard disk data of the invaded UNIX machines so as to create forensic image and carry out the methods and the specific steps of forensic analysis.

Ⅱ. THE INITIAL RESPONSE OF UNIX SYSTEMS

A. Create the initial response toolkit

In order to make initial response to the UNIX system, we should firstly prepare a CD-ROM or floppy disk equipped with the following tools (make sure that these tools are absolutely clean):

ls	dd	des	file	pkginfo
find	icat	lsof	md5sum	netcat 或 cryptcat
netstat	pcat	perl	ps	strace
strings	truss	df	vi	cat
more	gzip	last	w	rm
script	bash	modinfo	ismod	ifconfig

B. Initial response information preservation

The initial response information can be saved in the following ways:

1) Save data on the local hard disk;

2) Save data in remote media such as floppy disks, USB drives or tape drives;

3) Manually record information;

4) Use netcat (or cryptcat) command, and transfer the located data to the forensic analytical engine through the network.

Try not to save the data on the local hard disk. When we do data recovery or forensic analysis, the data stored on the local hard disk would overwrite the deleted data located in unallocated spaces, for these data may provide evidence while investigating. It is recommended to use netcat command and the network to transfer data to the USB drive or storage device of forensic analysis machine with enough space.

C. Data collection

Initial response phase should collect at least the following data:

a) System date and time;

b) List of the currently logged in users;

c) Time / date stamp of the whole file system;

d) List of currently running processes;

e) List of currently open sockets;

f) Applications monitoring on the open socket;

g) System list currently or recently connected to the system.

1) Execute the trusted shell

The first step of all responses is to make sure that the executing command is a trusted shell command. The attacker can implant in a Trojan horse program in an UNIX shell to record all executed commands, or perform some malicious actions that are imperceptible for investigators. Therefore, we must execute the credible shell.

First of all, we use the following command to load the credible Toolkit (suppose trusted kit is on a floppy disk) into the directory "/ mnt / floppy":

mount /dev/fd0 /mnt/floppy

Then enter the /mnt/floppy directory, input the following command to execute our credible shell:

#. / bash

Then we can perform the following credible orders.

2) The execution of command "date"

The execution of command "date" can record the time and date of the system. The local date and time setting is associated closely to the time / date stamp behind, they can also display the time you are in the system.

3) The execution of command "ifconfig"

The execution of command "ifconfig" can help us to get the information of each network card, including the network address and status. By analyzing these data we can find out that whether the intruders modified the IP address or started network monitoring program. Command format is as following:

ifconfig -a

4) Execute the "ps" command

By executing the "ps" command we can find out the name, command line parameter, the running time of each process, and the user's information that called from the process. Thus we can find out malicious processes. Command format is as follows:

ps -aux

5) Run the "netstat" command

By running the "netstat" command we can obtain the information of the open network sockets and others'. Under normal circumstances, the intruder often leave backdoor component on the compromised system. We can find out the backdoor component by analyzing the open network sockets. Command format is as follows:

netstat –an

6)Execute the "w" command

By executing the "w" command, it shows the information of the currently logged in users. For example, we can find out the logged in users' ID and the system they currently log in from, the operations that are running in the system, and also the date and system time.

7) Execute the "ls" command

The same as windows systems, each file and directory in UNIX systems has three time / date stamp to be collected: access time (atime), modification time (mtime) and the inode change time (ctime). We can use the trusted "ls" command wearing the appropriate command line parameters to get the times. The following command lines will tell you how to obtain the time / date stamp, and store the output results in a trusted floppy disk.

ls – alRu / > /floppy/atime

ls – alRc / > /floppy/ctime

ls – alR / > /floppy/mtime

III. FORENSIC ANALYSIS OF UNIX SYSTEM

A. Data acquisition

When we obtain the hard disk data of the victim machine, we shall obtain all the data on the hard disk including the data in the undistributed space. Only in this way that we can recover all the deleted files. By using the "dd" tools in UNIX system, we can finish this work. Now, we call all the backup data on the hard drive as forensics image. All the forensic analysis works are carried out based on forensic image instead of in the original disk.

Generally, we can use three methods to acquire the hard disk data:

a) Use "netcat" tools and the network. Namely, connect forensic analysis device with the victim machine using network, and the "netcat" tools are used for data transmission;

b) Take off the victim machine's hard drive, and install it onto a credible forensic analysis device;

c) Use a reliable operation system to start the victim machine directly, and copy the hard disk data to an external hard drive.

1) Equipment identification

The first step of obtaining data is to identify the name of the hard disk. In UNIX system, all the ATA/IDE device names of ATA/IDE hard disks are /dev/hd? While all the SCSI device names are /dev/sd?. "?" indicates the hard disk number can be replaced by English letters, such as /dev/hda, /dev/hdb, etc. We adopt the following commands to identify the system hard disk:

dmesg | grep –e [hs]d

2) Data acquisition

If you have took off the hard drive of the victim machine, you can install it into a credible evidence analysis machine, or if you have used a credible operating system to restart the victim machine, then you can get the data according to the following steps:

a) Determine the source disk and the destination disk to store source image. Here we use SRC to represent the source disk and DST to represent the destination disk.

b) Install the destination disk:

mount /dev/DST /mnt

c) Calculate the hash value of the source disk:

dd if=/dev/SRC bs = 2048 | md5sum

Save the calculated results for later use.

d) Copy the entire contents of the source disk to the destination disk, and save them in a file:

dd if = / dev/SRC bs = 2048 of = /mnt/disk1. dd

e) Compute the hash value of the result file on the destination disk:

md5sum /mnt/disk1. dd

Comparing this value with the hash value made by step (3), if the two matches, it means the copy succeeds. Or else, the copy fails and goes to step (4) to do the copy again.

3) Image file decomposition

Because the analysis objects of most existing forensic tools are hard drive partitions instead of the entire hard disk, so when successfully get the forensics image, we must separate the partitions one by one. Before performing the separation, we must understand the position and length of each partition firstly. And there are two methods we can use:

a) Use the “fdisk” tools in UNIX

```
# fdisk -lu disk1.dd

Disk disk1.dd: 0 heads, 0 sectors, 0 cylinders
Units = sectors of 1 * 512 bytes

Device    Boot    Start       End    Blocks   Id  System
disk1.dd1   *        63    208844    104391   83  Linux
disk1.dd2        208845   2249099   1020127+  83  Linux
disk1.dd3       2249100   4289354   1020127+  83  Linux
disk1.dd4       4289355  39873329  17791987+   5  Extended
disk1.dd5       4289418   4819499    265041   82  Linux swap
disk1.dd6       4819563  39873329  17526883+  83  Linux
```

In the above commands, the parameter "- l" indicates the position and length of the listed partition, the parameter "u" indicates that the unit of position and length is sector.

b) Use the “mmls” tools in the Sleuth Kit

```
# mmls -t dos disk1.dd
DOS Partition Table
Units are in 512-byte sectors

     Slot  Start      End        Length     Description
00:  -----  0000000000 0000000000 0000000001 Primary Table
01:  -----  0000000001 0000000062 0000000062 Unallocated
02:  00:00  0000000063 0000208844 0000208782 Linux (0x83)
03:  00:01  0000208845 0002249099 0002040255 Linux (0x83)
04:  00:02  0002249100 0004289354 0002040255 Linux (0x83)
05:  00:03  0004289355 0039873329 0035583975 Extended (0x05)
06:  -----  0004289355 0004289355 0000000001 Extended Table
07:  -----  0004289356 0004289417 0000000062 Unallocated
08:  01:00  0004289418 0004819499 0000530082 Linux Swap (0x82)
09:  01:01  0004819500 0039873329 0035053830 Extended (0x05)
10:  -----  0004819500 0004819500 0000000001 Extended Table
11:  -----  0004819501 0004819562 0000000062 Unallocated
12:  02:00  0004819563 0039873329 0035053767 Linux (0x83)
```

In the above commands, the parameter "-t dos" specify the partition type. The advantages of “mmls” tool are that it not only can list the location and the length of each partition, but also lists the location and length of the unallocated spaces. The unit of location and length of each partition is also sector.

When we understand the location and length of each partition, we can use “dd” utility to extract the contents of each partition from the image file:

```
# dd if=disk1.dd skip=63 count=208782 of=hda1.dd
# dd if=disk1.dd skip=208845 count=2040255 of=hda2.dd
```

The above command extracts contents of the first two partitions from the image file disk1.dd, respectively store them in file hda1.dd and hda2.dd.

B. Forensic analysis

After the data acquisition is completed, the next job is forensic analysis. Forensic analysis shall be carried out based on the acquired image file.

1) Preparation

Before forensic analysis, we install the acquired partition image file into the forensic analysis machine. Suppose we install hda1.dd in the / home/user01/analysis directory on the forensic analysis machine:

#mount –o ro,loop,nodev,noexec hda1.dd /home/user01/analysis

Then, enter /home/user01/analysis directory to start the analysis.

2) Forensic analysis

a) Locate the hidden files

During invasion process invaders often create some files that cannot be found easily by users, therefore, the first step of the forensic analysis is to find out the hidden files in the forensics images. After found the hidden files, we can infer the intruders’ intents by looking into the contents. Intruders often use a hiding technology to add ordinary files to the / dev directory. We can use the “find” command with "type-f" option to find ordinary files in the / dev directory.

Another hiding method is to change the first letters of the file name into “.”, because ordinary “ls” command cannot list the files that starts with “.”. However, we can use the “find” command plus "- name" to find out all the files that start with “.”.

b) Hidden files analysis

After finding the hidden files, we must analyze its contents to infer the intruders’ intrusion. We can use the “strings” command in UNIX systems (the main function of the command is to extract the ASCII string in the file) or Autopsy tools to analyze the contents of the hidden files.

c) Analyze the startup files and configuration files and etc.

Most intruders hide the backdoor program in the system configuration files and startup files. So, whenever the system starts, the backdoor program will automatically execute. Therefore, we can find out the suspicious programs by analyzing the system startup and configuration files.

As many erection sequences add a few commands behind the startup file, so as long as we check the last few lines of command in the startup files, we can notice the suspicious program. Normal commands in the startup files are generally included in an “if” statement structure, those commands without “if” statement structure can generally be identified as

suspicious programs. But it is not absolutely right, because intruders intend to fake things.

By analyzing the configuration files, we can also find out many useful invasion clues. Configuration file /etc/passwd contains all the login information of all system users, and intruder might create a new user or account in the system. Analyze the contents of file /etc/passwd, we may find some suspicious user information. Especially the users whose UID are 0 are mostly doubtful, because they are the root user with high permissions.

In addition to the startup files and configuration files, historical files can also provide some clues. Historical files exist in the users' / home directory (such as the file .bash_history), they record all the commands that users executed previously. After invaded the system, invaders often delete the files or link them to /dev/null. By restoring the files or analyze /dev/null, we can find out many useful clues.

We can find some quite useful invasion clues from the log files. UNIX system uses syslog daemon to establish the system log. Application programs or other hosts on the network send information to a daemon, and then daemon store the information in a log file. Before analyzing the log files,we firstly need to check the log configuration file / etc / syslog.conf. The file states the path of the log file. Under normal circumstances, log files are stored in / var / adm or / var / log. Here are some log files in directory / var / log:

boot: This log file stores system startup information, its content have preferable readability.

lastlog: This log file stores each user's last login information, including login time and location. If an intruder steals from other users, we can notice from the file. The file is a binary file.

maillog: The file stores log information of the mail system.

messages: This log file stores log information of most applications, usually including the start task information, login information. And also, it shows who tried to log in to the system to get the super administrator privileges. Major invasion clues can be found in the file, but the file is easily to be deleted or modified.

secure: To record the time and place that all the users logged in and they means they used after the system started.

wtmp / wtmpx: to store history information of entered users. It saved all the system login and exit information, and system startup and shutdown record, pursuant to which we can find some useful clues to the invasion. The file is a binary file, so we can read out the contents of the wtmp file if we use the "last" command with the "-f" option.

After the file is deleted, the file contents are still saved on the data blocks on the hard disk, but the system has been identified the data blocks as free. Then, we can find crime evidence in free data blocks.

We can use "dls" tool in The Sleuth Kit to extract data in the free data blocks. This "dls" tool would check the status of each data block, and output the data in the free data blocks.

After extracted the data from the free space, we can use the "foremost" tools (http://foremost.sourceforge.net) to analyze them. Foremost tools could find out specific file header and file footer from these data, consequently, files can be recovered.

We could also look for crime evidence from the swap space. Although there is no dedicated forensic tools to analyze the swap space, but we can use the "strings" command. Shell history, environment variables, process memory and other information can be found from the swap space.

Ⅳ. Conclusions

As UNIX is one of the most common mainstream operating systems, it is quite important to analysis the forensic method of UNIX system. This article discusses a preliminary study on forensic analysis of UNIX system. Interested readers could do further explorations and researches in this regard.

References

[1] Kevin Mandia,Chris Prosise,Matt Pepe.Incident Response 2nd. The McGraw-Hill Companies,Inc. 2003

[2] Brian Carrier. UNIX Computer Forensics. http://searchenterpriselinux.techtarget.com/searchEnterpriseLinux/downloads/Honeynet_Ch12.pdf.2007

[3] Keith J. Jones. Performing Investigations on a Live Host. http://www.usenix.org/publications/login/2001-11/pdfs/jones.pdf. 2007