Computer Forensics – Tracking the Cyber vandals

Martin Khoo, CISSP
Assistant Director, Defence & Response
Head, SingCERT
Infocomm Development Authority of Singapore (IDA)
Scope

♦ What is an IR Toolkit?
♦ Setting up a Forensic Analysis Workbench
♦ Forensic Process
♦ Tools of the trade
♦ Case Study
♦ Forensic Failure & Anti-Forensics
So you have an incident………

♦ Investigator needs to know
  – where to look for clues
  – how to obtain necessary information from the victim system

♦ Often times the search path is not very clear and the plain sight information is not sufficient

♦ Be mindful that the information may be needed for criminal proceedings
What is an IR Toolkit?

♦ A carpenter or mechanic has a toolkit to do their work; a surgeon has a surgical kit to perform open heart surgery.

♦ IR Toolkit is used by incident responder to assist them in carrying out their mission – investigate the cause of an incident.

♦ Consists of a mixture of tools and utilities; most of them are system/platform specific because it has to be customised to the particular operating system.

♦ Most of them are public domain and are pretty useful to the investigator.

♦ Some develop their own for specific purposes.
Forensic Analysis Workbench

♦ Forensic Analysis Workbench (FAW)
  – Hardware
    • Processor – PIII if you don’t like waiting
    • RAM (min -128 MB )
    • disk space – as much as you can (rule of thumb at least equal to the largest hdisk you have)
    • SCSI card
    • Tape Drive and media – DDS2/3
    • CD Writer
  – Software
    • system tools (clean copy) – statically linked
    • Freeware - The Coroner’s Toolkit (TCT) and related tools
    • Commercial - Encase Professional
Assembling your toolkit

♦ Assemble your IR kit and put it on a CD
♦ This will prevent the tools themselves getting trojaned, deleted or infected with viruses
♦ Also essential to have a scripting tool such as Perl on the CD
♦ An important resource which many investigators do not have in their IR Kit
♦ Use it to parse collected information and as a glue language for combining and parsing commands
♦ Can also use it to create utilities that provide needed functionalities not found in other tools
Preparing for Autopsy

Absolutely essential rules of forensic analysis

- Do NOT work on the actual system
- Make an image copy of the system’s hard drive and work on the image

♦ If circumstances do not permit taking an image dump then be careful when working on the victim system so as not to contaminate the information on the system

♦ This is very IMPORTANT if there is a possibility that the case will turn into a criminal investigation

♦ Caveat: Collecting volatile information may require that you work on the victim system
Image Copy

- There are tools available to do this – commercial and freeware
  - Encase Professional
  - Symantec - Norton Ghost
  - PowerQuest – DriveImage, DriveCopy
  - Freeware – “dd” utility

- Hardware disk imaging tool
  - Fastbloc (Encase)
  - AVCDEF (Vogon)

- Caveat: The tool must not alter the timestamp of the files on the original copy (victim system)
Forensic Process

♦ Acquisition
  – Get the data (normally disk images of the compromised system)

♦ Reconnaissance
  – Use a variety of tools to capture both volatile (if working on a live system) and non-volatile (disk related) information
Forensic Process

♦ Recovery
  – Attempt to get back “evidence” that has been removed (although not completely)
  – Typically these are disk-related information like files and directories

♦ Time Lining
  – Chronology of events; who did what when and hopefully also how it was done
Phases

- Forensic investigation of a compromised hard drive has 2 major phases
  - Collection/acquisition of data
  - Analysis of collected data
- Phase I can be accomplished using tools mentioned in previous slide
Type of Information

♦ Volatile Information
  – Certain state information of the system is transient
    • Network connections
    • Routing table
    • Process Table
    • Open ports
    • Users
  – These information are critical to the investigation and should be recorded as soon as the incident is discovered
  – Need tools to assist in capturing these information
  – Some of the tools can be system tools provide by the operating system
  – Third-party freeware tools can be used to get to the rest
Volatile Information

♦ Network Information
  – Actual communications between victim system and other systems

♦ Active Process List
  – programs/daemons active on the victim system

♦ Logged-in user list
  – who is using the victim system

♦ Open files, libraries in use
  – what files (hidden) or trojans (rootkit) are loaded in the victim system
Non-volatile Information

♦ Information that remains after a reboot
  – configuration settings
  – System and data files
  – Registry settings

♦ Can be accessed through mapping of drives from the victim system to the FAW

♦ If possible work on an image copy of the victim system
Non-volatile Information – MAC (1)

♦ MAC Times

– Files on the victim system will potentially give important clues to the investigator; not only from their content but certain attributes
– In particular the investigator should record 3 important time attributes of the files
  • Modified
  • Access
  • Creation
– Collectively referred to as MAC times
– MAC times are one of the most valuable information to an investigator
Non-volatile Information – MAC (2)

♦ How do we get the MAC times?
♦ Careful not to inadvertently change those attributes while trying to determine their values
♦ There are tools that can assist in the culling of these information
  – Foundstone Forensic Toolkit - Afind
  – Mac.pl – Perl script to obtain the MACs in a comma-delimited output format (can use it in Excel for further processing)
♦ Once the MAC times are obtained and recorded the investigator may then proceed to do perform other checks on the victim system
Non-volatile Information – MAC(3)

♦ Note that calculating checksums or doing file signature analysis will alter the MAC times on the files
♦ Always do the MAC time capture BEFORE you do anything else to the files on the victim machine
♦ You should then do a MD5 checksum on the MAC time output file to ensure integrity is maintained
♦ Be mindful of chain-of-custody procedure
Non-volatile Information – LOG (1)

♦ System/Application Logs
  – The trustworthiness of the system/application logs depends on the type of incident
  – If privilege access is obtained by the intruder then the integrity of the logs is in doubt
  – The logs of interest to an investigator
    • Event Logs (System, Security, Application)
    • Specific application logs (IIS, SQL Server etc)
  – Tools available to dump the NT event logs for further processing (dumpevt.exe, dumpevt.pl)
Non-volatile Information – Strings

♦ Discovering what an unknown executable is trying to do

♦ Most programs will have ASCII strings in their code. These strings remain intact when the code is compiled

♦ Investigator is looking for:
  – Usage statement
  – Author
  – System/Function calls

♦ Can infer what the program will do when executed

♦ Tools: Strings.exe, Finfo.pl
You have data....

- Phase II is when you need to derive information out of the collected data
- As in Phase I, tools are needed to assist the investigator
- We will look at 3 major open source forensic tools
- They complement each other
Tools of the Trade

- Tools are platform specific and are mainly freeware contributed by the security community
  - windows specific tools consists of collection of Perl scripts
  - the flag bearer for Unix/Linux is a collection of tools known as The Coroner’s Toolkit (TCT) and 2 other complementing utilities
    - TCTUtils
    - Autopsy Forensic Browser
Bring in the coroner…

♦ TCT – The Coroner’s Toolkit
  – Collection of 4 distinct group of tools
  – Provide powerful techniques for collection and analysis of forensic data
  – Goal is to reconstruct past events and/or recover deleted data
  – Can analyse activities on a live system and capture state information
Coroner’s Toolkit (1)

♦ Grave-robber [Acquisition]
  – Captures various types of data and creates MD5 hashes of them to preserve integrity
  – Information captured include the (M)odified, (A)ccess and (c)hanged times, deleted files that are still open, memory contents
  – MAC time can be changed and also if too much time have elapsed, data related to the attack may be overwritten
Coroner’s Toolkit (2)

- Pcat, ils, icat, file [Reconnaissance]
  - Records and analyses processes and filesystem (inode) data
  - pcat – copies process memory from a live system
  - ils – list inode information
  - icat – copies files by inode number
  - File – classifies files into various types
Coroner’s Toolkit (3)

♦ Mactime [Timelining]
  – Creates a chronological timeline of when files have been Modified, Accessed, or Changed (MAC) for each inode, along with their filenames
Coroner’s Toolkit (4)

♦ unrm, lazarus [Recovery]
  – Recovers and analyses unallocated disk blocks on a file system
  – unrm – collects information in unallocated portion of the file system
  – lazarus – analyses raw data from unrm and attempts to classify the data
Extending the TCT (1)

♦ TCTutils
   – Collection of utilities that adds functionality to the TCT
     • bcat – display contents of disk block to stdout
     • blockcalc – map dd images and unrm results
     • fls – display file and directory entries that have been deleted (use with –d option will list the names of all the deleted files on the disk)
     • find_file – determine which file has allocated an inode
Extending the TCT (2)

- `find_inode` – determines which inode has allocated a disk block
- `Istat` – display information about an inode
- `mac_merge` – merges the output from “fls -m” with the output from “mactime” to create a consolidated timeline file
Forensic Browser

♦ GUI front end to TCT and TCTUtils
♦ Allows an investigator to browse and analyse forensic images at the file, block, and inode level
♦ Allow for searching for keywords
♦ Approaching the capability of forensic tools on Windows (e.g. Encase)
Forensic Case Study
Overview of Process

- Evidence
- Acquisition & Reconnaissance
- Analysis
- Recovery

-grave-robber
-ils
-il2mac
-fls -m

- Timelining
- AFB
- lazarus

-icat
-unrm
Evidence Collection

- Evidence is acquired in the form of “dd” images of each of the partition of the hard disk.

- Syntax of command used
  - `dd if=/dev/hda1 of=/forensic/images/hda1.dd`

- You will also want to calculate checksums on the images to ensure that their integrity can be verified (MD5)
Preparing for Autopsy

♦ Mount the captured images on the forensic machine with non-intrusive options
  – Read only
  – No execute

♦ This is to prevent altering the information during the forensic examination process
Preparing for Autopsy

♦ The layout of the victim hard disk is as follows
  – / - hda8.dd
  – /boot – hda1.dd
  – /usr – hda5.dd
  – /home – hda6.dd
  – /var – hda7.dd
Preparing for Autopsy

♦ `mount -o ro,loop,nodev,noexec images/hda8.dd mnt`
♦ `mount -o ro,loop,nodev,noexec images/hda1.dd mnt/boot`
♦ `mount -o ro,loop,nodev,noexec images/hda5.dd mnt/usr`
♦ `mount -o ro,loop,nodev,noexec images/hda6.dd mnt/home`
♦ `mount -o ro,loop,nodev,noexec images/hda7.dd mnt/var`
Acquisition & Reconnaissance Phase

♦ We use a combination of TCT and TCTUTILS to accomplish this

♦ What this phase accomplishes are
  – Obtain the MACs for existing files (grave-robber)
  – Obtain the MACs of unallocated inodes (ils)
  – Convert the ils output to same format as “grave-robber” (ils2mac)
  – Collect MACs of recently deleted files (for Linux only) (fls –m)
Deleted Files (1)

- Two types of data are deleted when a file is deleted from the file system
  - (i) Inode value in the directory entry
  - (ii) Inode content such as file size and block pointers
- If (i) is deleted then only the filename can be recovered
- If (ii) is deleted then the MACs can be recovered
Evidence Processing Scripts

# Script to process images using ils and ils2mac

```
for i in 1 5 6 7 8
do
  ils images/hda$i.dd | ils2mac > data/hda$i.ils
done
cat data/hda?.ils data/body.ils
```

# Script to process images using fls

```
fls -m "mnt/boot/" images/hda1.dd > data/hdal.fls
fls -m "mnt/usr/" images/hda5.dd > data/hda5.fls
fls -m "mnt/home/" images/hda6.dd > data/hda6.fls
fls -m "mnt/var/" images/hda7.dd > data/hda7.fls
fls -m "mnt/" images/hda8.dd > data/hda8.fls
cat data/hda?.fls data/body.fls
```
Deleted Files (2)

♦ If nothing is deleted then, one can recover all or part of a file (good!)
♦ Different OS deletes one, both or none of these information
♦ Linux – saves both
♦ OpenBSD – saves (i) deletes (ii)
♦ Solaris – deletes both (tough!)
Timelining Phase

♦ The output from the following are concatenated
  – Grave-robber
  – ils2mac
  – fls –m

♦ The output becomes the input to “mactime”
  – mactime –p mnt/etc/passwd –g mnt/etc/group –b
data/body.all 11/07/2000 > data/mactime.txt
Initial Analysis

♦ Now comes the crucial part – analysing the output of “mactime”
♦ You are looking for something suspicious; out of the norm

– Hints
  • Directory(ies) with unusual name(s)
  • File(s) with unusual names
  • Execution of certain commands
  • Keywords
Nov 08 00 22:51:56 156 ..c -rwxr-xr-x 1010 users
/mnt/hdbpart/4/forensic/chal/mnt/usr/man/.Ci/needz
4096 ..c drwxr-xr-x 1010 users
/mnt/hdbpart/4/forensic/chal/mnt/usr/man/.Ci/paki
1052024 ..c -rwxr-xr-x 1010 users
/mnt/hdbpart/4/forensic/chal/mnt/usr/man/.Ci/bx
12408 ..c -rwxr-xr-x 1010 users
/mnt/hdbpart/4/forensic/chal/mnt/usr/man/.Ci/addn
118 ..c -rwxr-xr-x 1010 users
/mnt/hdbpart/4/forensic/chal/mnt/usr/man/.Ci/ /Anap
8524 ..c -rwxr-xr-x 1010 users
/mnt/hdbpart/4/forensic/chal/mnt/usr/man/.Ci/paki/slice2
18535 ..c -rwxr-xr-x 1010 users
/mnt/hdbpart/4/forensic/chal/mnt/usr/man/.Ci/fix
83 ..c -rwxr-xr-x 1010 users
/mnt/hdbpart/4/forensic/chal/mnt/usr/man/.Ci/addps
328 ..c -rwxr-xr-x 1010 users
/mnt/hdbpart/4/forensic/chal/mnt/usr/man/.Ci/do
4096 ..c drwxr-xr-x 1010 users
/mnt/hdbpart/4/forensic/chal/mnt/usr/man/.Ci/
6793 ..c -rw-r--r-- 1010 users
/mnt/hdbpart/4/forensic/chal/mnt/usr/man/.Ci/paki/stream.c
699 ..c -rwxr-xr-x 1010 users
/mnt/hdbpart/4/forensic/chal/mnt/usr/man/.Ci/chmod-it
185988 ..c -rwxr-xr-x 1010 users
/mnt/hdbpart/4/forensic/chal/mnt/usr/man/.Ci/find
188 ..c -rwxr-xr-x 1010 users
Nov 08 00 22:52:09 9 m.c lrwxrwxrwx root root
/mnt/hdbpart/4/forensic/chal/mnt/.bash_history -> /dev/null
9 m.c lrwxrwxrwx root root
/mnt/hdbpart/4/forensic/chal/mnt/root/.bash_history -> /dev/null
Recovery Phase

♦ The main tool here is the Autopsy Forensic Browser (AFB)
♦ It allows an investigator to perform the following functions
  – File & Directory Browsing
  – Block Browsing
  – Inode Browsing
  – Block Searching
Autopsy Forensics Browser
ver 1.01

Browsing Type Images
Files: /mnt/hdbpart/4/forensic/chef/nuts/honeypot.hda8.dd
Block: /mnt/hdbpart/4/forensic/chef/nuts/boot/honeypot.hda1.dd
Node: /mnt/hdbpart/4/forensic/chef/nuts/home/honeypot.hda5.dd
Search: /mnt/hdbpart/4/forensic/chef/nuts/usr/honeypot.hda5.dd

Time Zone difference: 0 hours

Enter the Lab

Help
File Browsing

- Use the directories/files identified in the “mactime” output as the source
- Use a graphical tool to perform browsing of the identified suspicious directory (/usr/man/.Ci)
- The purpose of this step is to accomplish 2 objectives:
  - View the contents of file(s) in the directory (both existing and also deleted ones)
  - Attempt to recover deleted file(s)
Detail Analysis

♦ Some interesting deleted files that can be noticed includes named.tgz, named.tar, ssh-1.2.27.tar, nfs-utils-0.1.9.1-1.i386.rpm and wuftpdp.rpm

♦ These files can be recovered using the icat utility of TCT

♦ Need to specify the inode of the required file (obtain from the AFB listing)
File Recovery

- We will attempt to recover two of the deleted files using icat.
  - Syntax: `icat <image> <inode> > <save file>`
    - `icat images/hda5.dd 109791 > recovered/hda5.ssh-1.2.27.tar`
    - `icat images/hda5.dd 109861 > recovered/hda5.named.tar`
    - `icat images/hda5.dd 109866 > recovered/hda5.wuftpd.rpm`
<table>
<thead>
<tr>
<th>File</th>
<th>Last Modified</th>
<th>Size</th>
<th>Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>r</td>
<td>05/03/2000</td>
<td>552B</td>
<td>100 101</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ses-1.2.27.tar</td>
<td>08/10/2000</td>
<td>18698240</td>
<td>100 101</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>syslogd</td>
<td>06/03/2000</td>
<td>359996</td>
<td>100 101</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>temp2</td>
<td>10/30/1996</td>
<td>2286B</td>
<td>20 17275</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>meteo</td>
<td>06/03/2000</td>
<td>147900</td>
<td>100 101</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>temp3</td>
<td>10/30/1996</td>
<td>2414B</td>
<td>20 17275</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>temp4</td>
<td>10/30/1996</td>
<td>3063B</td>
<td>20 17275</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>temp5</td>
<td>10/30/1996</td>
<td>2843B</td>
<td>20 17275</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>clean</td>
<td>06/03/2000</td>
<td>698B</td>
<td>100 101</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>snap</td>
<td>06/03/2000</td>
<td>3098B</td>
<td>100 101</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ptyp</td>
<td>10/30/1996</td>
<td>2213B</td>
<td>20 17275</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>q</td>
<td>05/03/2000</td>
<td>133344</td>
<td>100 101</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>killall</td>
<td>05/03/2000</td>
<td>12495B</td>
<td>100 101</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>perror</td>
<td>05/03/2000</td>
<td>49800B</td>
<td>100 101</td>
</tr>
<tr>
<td>* r</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fix</td>
<td>05/03/2000</td>
<td>18533B</td>
<td>100 101</td>
</tr>
</tbody>
</table>

Contents of File: /mnt/hdbpart4/forensic/chal/mnt/usr on honeypot.hda5.dd

```c
; 20000 _mp_submul_1 -- Multiply a limb vector with a single limb and subtract the product from a second limb vector. 
; copyright (c) 1992, 1994 free software foundation, inc. 
; this file is part of the GNU MP Library. 
; the GNU MP Library is free software; you can redistribute it and/or modify 
; it under the terms of the GNU Library general public license as published by 
; the Free Software Foundation; either version 2 of the License, or (at your 
; option) any later version. 
; the GNU Library is distributed in the hope that it will be useful, but 
; without any warranty; without even the implied warranty of merchantability 
; and fitness for a particular purpose. see the GNU Library general public license 
```
File Analysis

- AFB also allows an investigator to do a “strings” on binary files
- “Strings” is used to find out what are the ASCII strings embedded in an unknown executable
- The output can be used to infer the functionalities of the executable
Reporting

♦ AFB also provides the ability to generate a report containing the strings output and all the necessary attributes of a binary
  – UID, GID, inode, MACs, disk blocks allocated, MD5 checksums
♦ This will form part of the forensic report that an investigator would need to produce
Strings Output

/lib/ld-linux.so.2
__gmon_start__
libc.so.6
strncpy
tclsh
utimes
system
settimeofday
shmctl
strocr
fprintfderegister_frame_info
fseek
ferror
strncpy
_x0_gete
freed
gettimeofday
ftell
fprintf
fclose
stderr
fwrite
exit
fopen
_x0_stdin_used
__libc_start_main
__register_frame_info
_strerror
errno 2.0
errno 2.1
fflush
ev catalogue cannot open
fix: Can't open %
fix: Last 17 bytes not zero
fix: Input data checksum 0

-00
And so………

♦ TCTUTILS and the Autopsy Forensic Browser provides many critical functions that complement that of the TCT
♦ The AFB provides a GUI front end to make the low-level forensic tools in the TCT and TCTUTILS easier to use
♦ However, the investigator has to come up with his/her analysis and conclusion
Forensic Failure & Anti-Forensic
Forensic Failure

♦ Compromise is at the kernel space instead of user space
♦ Traditional forensic tools completely failed during an investigation
♦ Change the behavior of ANY command executed on the system without changing the program binaries themselves
♦ Any trusted toolkit an investigator used will be useless in such a situation
Loadable Kernel Module

♦ LKM provides dynamic functionalities by altering a running kernel without rebooting
  – Linux - insmod <module name>
♦ LKM can access all functions and memory areas of a kernel
♦ A rouge LKM will create untold havoc
LKM gone awry….

♦ LKM intercepts system calls made by user space programs
♦ The “syscall” table is altered which changes the OS behavior
♦ “syscall” table is the entry point into the OS provided to user space programs and lives in kernel space
Evil LKM

♦ OS maps all user space commands to the function calls on the syscall table

♦ KNARK (a evil Linux LKM) alters the following system calls on the syscall table
  – Getdents – get directory entries
    • By compromising this call, knark is able to hide files and directories from user-level program
Evil LKM

- **kill** – sends a signal to a process to kill it. Extra signals (#31) will trigger the option flags of a process to be set to the “hidden” state. Signal #32 unhides the process.
- **read** – reads content from a disk file. Knark compromises this system call to hide intruder connection specifics from `netstat`.
- **ioctl** – changes the behavior of files and devices. Use to clear the promiscuous flag on the network interface.
- **fork, clone** - spawns a new process. Use to hide all child process created from a hidden parent process.
So How……

♦ Preventive Measures
  – Securing the “syscall” table
    • Using a LKM (of course) to watch the syscall table and monitor when other LKM are loaded
    • This “sentry” module will alert the sysadmin when changes are made to the syscall table

♦ Investigative Measures
  – Kernel mode “ps-like” tool
    • Retrieves executable images of each process (Carbonite)
Anti-Forensics

- Attempt to downgrade the quantity and quality of the “evidence” that will be able to an investigator
- Data hiding and Data destruction
- Exploiting limitations of the current forensic tools (TCT, TCTUTILS)
- The anti-forensic techniques works for the ext2 file system (Linux)
Anti-forensic Toolkit

- The Runefs toolkit allows hiding of data in areas of the hard drive that the TCT will never look.
- The Defiler’s Toolkit (TDT) performs file system sanitization “with surgical precision.”
- Removes not only data blocks but also all entries in the file system pertaining to the existence of a file (inodes & directory entries).
Is there hope ........?

♦ Not to rely on the local file system as the only record of disk operations
♦ Storing inode updates to a remote and secure syslog server
♦ Requires modifying the kernel vfs layer to accomplish this
Resources

♦ The Coroner’s Toolkit (TCT)
  – http://www.porcupine.org/forensics

♦ TCTUTILS & Autopsy Forensic Browser (AFB)
  – http://www.cerias.purdue.edu/homes/carrier/forensics/

♦ The @stake Sleuth Kit (TASK)
  – http://www.atstake.com/research/tools/index.html#forensics

♦ TDT, Runefs
  – Phrack 59 – http://www.phrack.org
Thank You

markhoo@singcert.org.sg