Name: .....

Student ID: .....

LAB 4: WINDOWS FORENSICS

INTRODUCTION

In this lab, we will examine a forensic image taken of a computer that may have been used to send malware to a victim which allows the sender to gain full remote access to the target computer.

RESOURCES AND LINKS

- SANS Investigative Forensic Toolkit (SIFT) Workstation
  https://computer-forensics.sans.org/community/downloads
- The Sleuth Kit
  http://www.sleuthkit.org
- RegRipper
  http://regripper.wordpress.com/regripper/
- Windows Registry Hives
  http://msdn.microsoft.com/en-us/library/ms724877%28v=vs.85%29.aspx
- Internet Explorer Index.dat URL Records
  http://www.forensicswiki.org/wiki/Internet_Explorer_History_File_Format
- Digital Detective DCode
  http://www.digital-detective.co.uk/freetools/decode.asp

LAB OUTLINE

LAB 4: WINDOWS FORENSICS

LAB 4: WINDOWS FORENSICS

1. IMA HACKER REMOTE ACCESS TROJAN SCENARIO

In March, 2010, a warrant was executed on the residence of Imanuel Leet-Hacker (aka Ima Hacker), after police received reports of suspicious emails originating from his email address. Police seized a large amount of equipment from his home and, due to a large backlog, have been force to outsource some of the forensic work to you.

Emails originating from Ima Hacker reported to come from the FBI, and contained an attachment which, when run, attempted to open a connection to imahacker.no-ip.org

You have been asked to:

- Locate copies of the email sent to Somepoor Victim to prove that they did, in fact, come from Ima Hacker
- Locate the Trojan on Ima Hacker's computer, and verify that it is the same Trojan located by police
- Determine what tools (if any) were used in the creation of the Trojan

Any dates and times associated with the above files should also be determined.

The police have provided you with some information to assist your forensic examination.

EXHIBIT 1: SCREENSHOT OF EMAIL RECEIVED BY "SOMEPOOR VICTIM"

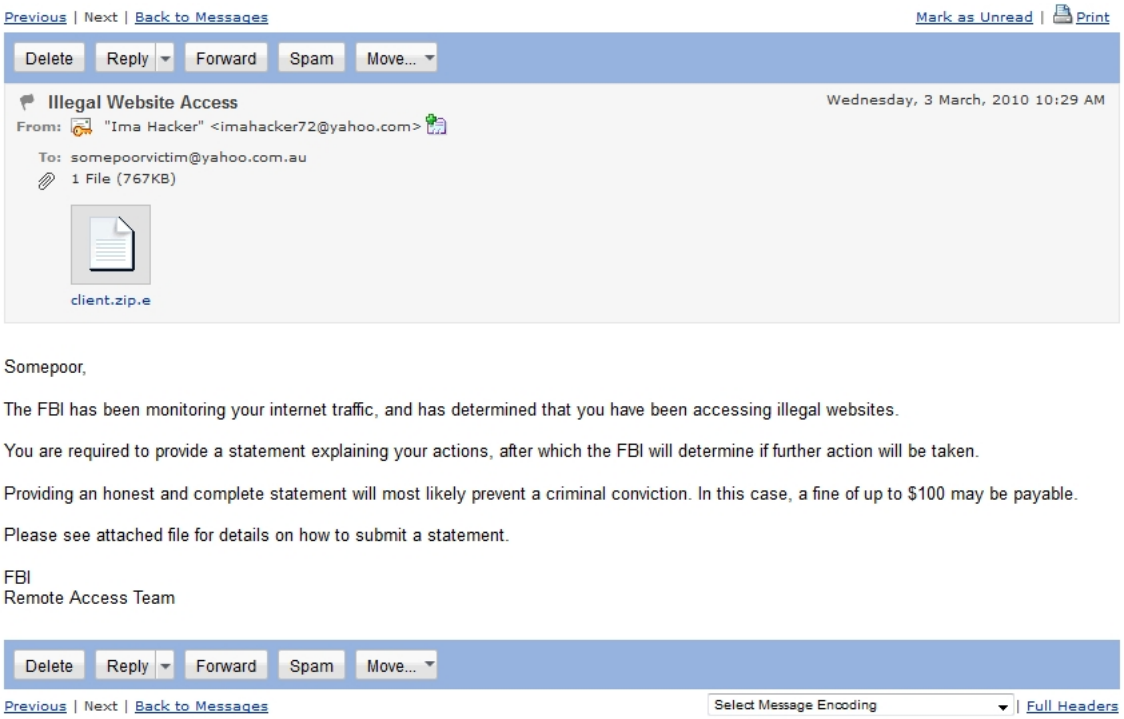


Figure 1. Screenshot of Email Message Received by "Somepoor Victim"

LAB 4: WINDOWS FORENSICS

EXHIBIT 2: EMAIL HEADERS OF EMAIL RECEIVED BY "SOMEPOOR VICTIM"

```
X-Apparently-To: somepoorvictim@yahoo.com.au via 98.138.85.213; Tue, 02 Mar 2010
16:30:00 -0700
Return-Path: <imahacker72@yahoo.com>
Received-SPF: none (mta1401.mail.mud.yahoo.com: domain of imahacker72@yahoo.com does
not designate permitted sender hosts)
X-Originating-IP: [98.139.91.56]
Authentication-Results: mta1401.mail.mud.yahoo.com from=yahoo.com; domainkeys=pass
(ok); from=yahoo.com; dkim=pass (ok)
Received: from 127.0.0.1 (HELO omp1056.mail.sp2.yahoo.com) (98.139.91.56) by
mta1401.mail.mud.yahoo.com with SMTP; Tue, 02 Mar 2010 16:30:00 -0700
Received: (qmail 93079 invoked by uid 1000); 2 Mar 2010 23:29:58 -0000
Received: (qmail 9062 invoked from network); 2 Mar 2010 23:29:56 -0000
Received: from testvm (imahacker72@220.233.43.115 with login) by
smtp145.mail.mud.yahoo.com with SMTP; 02 Mar 2010 16:29:27 -0700 PDT
Message-ID: <1D3DCC2B99B6465683CFC9D7CE938024@testvm>
From: This sender is DomainKeys verified
"Ima Hacker" <imahacker72@yahoo.com>
To: <somepoorvictim@yahoo.com.au>
Subject: Illegal Website Access
Date: Wed, 3 Mar 2010 10:29:20 +1100
MIME-Version: 1.0
Content-Type:       multipart/mixed; boundary="----
=_NextPart_000_000E_01CABABC.62D366C0"
X-Priority: 3
X-MSMail-Priority: Normal
X-Mailer: Microsoft Outlook Express 6.00.2900.5512
X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2900.5512
Content-Length: 1049806
```

Figure 2. Email Headers from Message Received by "Somepoor Victim"

EXHIBIT 3: DETAILS OF CLIENT.ZIP.EXE

File Name	File Size	MD5 Hash	SHA1 Hash
client.zip.exe	775168	9039c22d07eefd0e5dcaa92906a66fa2	a8594d66efcd8eb3346f7e662f685bb08eadb97f

Table 1. Details of Trojan File Attached to Email Message

LAB 4: WINDOWS FORENSICS

2. CASE SETUP AND IMAGE MOUNTING

For this lab, you'll need:

1. SIFT Workstation 3.0 (run directly from the ICT Virtual Machine Launcher)
2. Disk image "ImaHackerRAT.E01" (will be provided prior to the lab)
3. Email tool "undbx-0.21.tar.gz" (will be provided prior to the lab)
4. Dcode tool

WORKSTATION SETUP

1) Copy the "ImaHackerRAT.E01" image file and "undbx-0.21.tar.gz" tool and place them in the Host-Cases folder ("C:\apps\vm\ICT30010-SANS-SIFT\Host-Cases") on your windows machine (or your own alternate location if you are using SIFT from home).

2) Start SIFT Workstation

   a) Start the VMWare Launcher and locate the "SANS SIFT Workstation 3.0" from the ICT30010 Folder

   b) Once SIFT has started successfully log on as "sansforensics", with the password "*forensics*"

3) Prepare Lab Files

   a) Check that you can access the undbx-0.21.tar.gz and ImaHackerRAT.E01 file in the Host-Cases directory (/mnt/hgfs/Host-Cases) You can do this by listing files with the "ls" command and the "-l" option to display a "long" listing including file size (the file should be about 800mb), or use the GUI.

   b) Create a new directory for this lab: (/cases/lab4)

   c) Move the ImaHackerRAT.E01 file to this new case directory using the "mv" command (or via the GUI, if you prefer)

Linux Command Cheat Sheet:

ls	list files in the current directory
ls -al	list all files (including hidden) with additional details
cd *dir*	change directory to *dir*
cd	change to home
pwd	show current directory
mkdir *dir*	create a directory *dir*
rm *file*	delete *file*
rm -r *dir*	delete directory *dir*
cp *file1 file2*	copy *file1* to *file2*
mv *file1 file2*	rename or move *file1* to *file2*
more *file*	output the contents of *file*
head *file*	output the first 10 lines of file
tail *file*	output the last 10 lines of file
touch *file*	create or update the time on *file*
sudo *command*	run *command* as root
date	show the current date and time
man *cmd*	show the manual (help) for *cmd*
ping *host*	ping *host* and output results
ifconfig	show IP and MAC address info
df	show disk usage
fdisk -lu	list all disk partitions (run as root)
hdparm -I /dev/*sda*	show info about disk *sda*

LAB 4: WINDOWS FORENSICS

TIMEZONE SETTINGS

Times and dates are dependent on time zones, and some of the tools we're using today require that we know the time zone of the suspect's computer. We'll verify the Time Zone settings later, but for now, let's set our analysis machine's Time Zone to GMT+10 (Melbourne).

4) Change the Time and Date settings in Ubuntu/SIFT Workstation

   a) Click on the time at the top of the screen

   b) Select Time and Date Settings

   c) Click set manually, enter the correct date and time and set the location as Melbourne

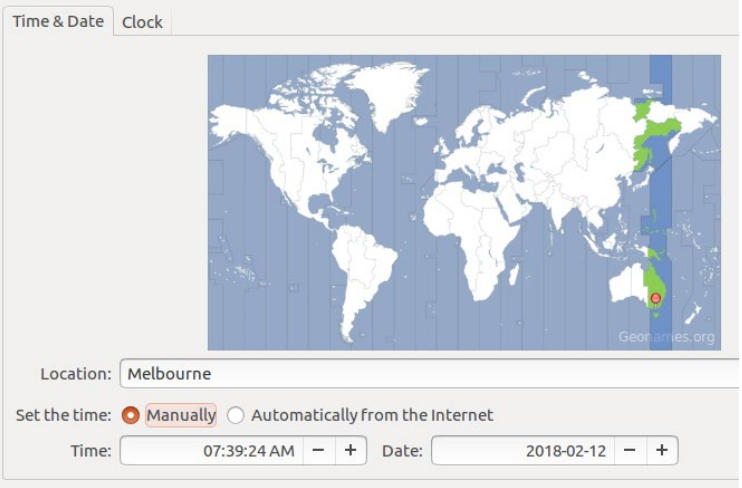


Figure 33. Time and Date Settings in SIFT Workstation

WORKSTATION SETUP

5) Often, we are faced with a situation where we need to install extra tools to support our forensic investigation. We will install an email extraction tool to assist with the extraction of email files

   a) Type the following command "`cd /mnt/hgfs/Host-Cases `".

   b) Type the following command "`tar zxf undbx-0.21.tar.gz`".

   c) Type the following command "`cd undbx-0.21`".

   d) Type the following command "`sudo ./configure`".

   e) Type the following command "`sudo make`".

   f) Type the following command "`sudo make install`

LAB 4: WINDOWS FORENSICS

IMAGE VERIFICATION

For this exercise, our disk image is not a standard "dd" (raw disk image). We have instead been provided with a compressed Expert Witness Format (EWF, also known as EnCase format, or E01) disk image. Because of this, a running md5 across the E01 file will *not* calculate the hash value of the enclosed disk image. One of the benefits of the EWF format is that the disk hashes are contained *within* the E01 file, and do not need to be supplied separately.

6) Let's see what information is contained within the E01 (other than the disk image, of course!). Run "ewfinfo" to check what information is available:

   a) Make sure you're in the "/cases/lab4" directory, you can type "pwd" to confirm the current path

   b) Run the command "ewfinfo ImaHackerRAT.E01"

      i) Note the sector size (bytes per sector) for the forensic image: ..... bytes

      ii) How large is the disk (media size)? ..... Gigabytes

      iii) Observe that the hash information is also included in the E01. In this case, both an MD5 *and* a SHA1 hash have been calculated.

      iv) The ewinfo command also pulls important metadata information regarding the acquisition of the forensic image, it is important to note items such as Operating System used and System date refer to the system that CREATED the forensic image, and NOT the system that the image is taken from.

      v) Acquisition Date: ……..

      vi) Operating System Used: ……

7) To verify the embedded hash, we run a purpose-built tool "ewfverify":

   a) Verify that the hash contained within the E01 file is correct by running "ewfverify ImaHackerRAT.E01" (this will take a minute or two)

DETERMINE PARTITIONS ON DISK IMAGE

We need to identify what partition(s) are on the forensic image, so let's examine the partition table using the Sleuth Kit tool "mmls" (you might remember it from last week).

8) Run the command "mmls ImaHackerRAT.E01". The output from this tool not only shows you the NTFS Partition, but also what other areas of the disk have been allocated for use (e.g. the first sector should be listed as containing the primary partition table)

a) Note the details of the partition reported by mmls for the NTFS partition (you will need a calculator to work out the total size). We established the sector size earlier using ewfinfo, but mmls also tells us how many bytes are in each sector. Total Sectors should be: EndSector-StartSector + 1 and Total Size in MB is (TotalSectors * SectorSize) / 1024 / 1024

Partition Number	Partition Type (e.g. NTFS/FAT)	Start Sector	End Sector	Total Sectors	Total Size (MB)
1					

Table 2. Partition Details for Forensic Image

MOUNT THE PARTITION

Mounting an E01 file is a little different than mounting a DD. First we have to gain access to the raw disk image contained within the outer E01 container, then we can mount the raw disk image as we usually do.

9) First, we need to access the raw disk image contained within the E01.

   a) Make the raw disk image within the E01 file available by running the ewfmount command: "sudo ewfmount ImaHackerRAT.E01 /mnt/ewf". (ensure you use sudo)

   b) Verify the mount occurred correctly, and see what files it has made available by getting a directory listing of the /mnt/ewf directory (run "sudo ls -l /mnt/ewf"), there should be a file named "ewf1"

10) Now we want to mount the partition contained within the raw disk image by determining the partition's offset from within the forensic image, then running the mount command.

   a) Calculate the starting offset in bytes (starting sector x sector size) from the values you noted earlier:

      Starting Offset: .....

   b) Mount the partition read-only by running
      "sudo mount -t ntfs -o ro,loop,offset=*xxxx*,show_sys_files /mnt/ewf/ewf1 /mnt/windows_mount"

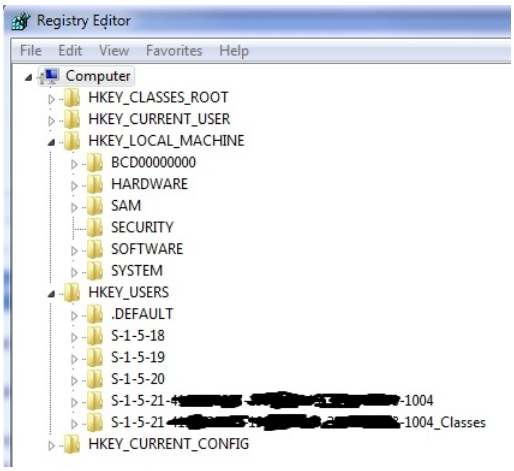
      (don't forget to replace *xxxx* with the offset you just calculated). The "show_sys_files" option makes sure the usually hidden "$" system files are visible (e.g. $Bitmap, $LogFile, etc.)

   c) Get a directory listing ("ls") of the /mnt/windows_mount directory to make sure the mount occurred successfully. (notice how all the $ files are visible from the show_sys_files option)

LAB 4: WINDOWS FORENSICS

3. MANUAL EXAMINATION

Let's try some quick manual searches to see if we can find anything interesting



11) Examine the Windows Registry settings to determine the current Time Zone

    The Windows registry contains settings and information relating to installed programs and hardware devices. Commonly viewed in Windows by using the "Registry Editor" program, these settings can also be viewed using forensic tools in SIFT Workstation.

    a) Most of the Windows registry files are located in the directory "C:\Windows\System32\Config". Find this directory under the mounted forensic image (/mnt/windows_mount) – run "cd /mnt/windows_mount/WINDOWS/system32/config/"

    b) Get a directory listing (ls), and look at the files located in this directory. Some of the most commonly examined files include:

        i) **SAM** is the Security Account Manager, and it is the registry hive which contains details about user accounts and the hashed passwords associated with those accounts.

        ii) **software** contains details about currently installed software and its configuration settings

        iii) **system** contains operating system level settings, including details of hardware devices and system settings such as the current time zone.

        iv) **AppEvent.Evt**, **SysEvent.Evt**, and **SecEvent.Evt** are the Windows event logs, which can contain many forensically useful pieces of information.

    c) RegRipper is a tool which extracts pre-defined registry keys, and creates a report for you.

        i) Run the RegRipper tool over the system registry hive by executing the command: "rip.pl -r system -f system > /cases/lab4/system.txt" (this will save the report to /cases/lab4/system.txt, if you re-run the same command it will give an error, try to delete the file fist with the rm command)

        ii) View the created report by typing "more /cases/lab4/system.txt". Under the "TimeZoneInformation key" heading, you should see a value for "StandardName" – this is the name of the current (non-daylight savings) time zone. What is the current "standard" time zone?

        .....

12) Now we're satisfied that we have the correct time zone, let's see if our suspect was silly enough to leave the Trojan file sitting on his computer. Let's see if we can find a file named "client.zip.exe"

    a) Change into the directory where we mounted the forensic image (cd /mnt/windows_mount)

    b) Run the command "find . | grep -i client.zip.exe" (find all files in the current directory and subdirectories, filter for the case insensitive name "client.zip.exe").

    c) Looks like we're not quite so lucky, no results returned. We'll get back to searching for this file later.

13) Since we're also looking for emails, let's see if there are any common types of email databases on his computer. Microsoft Outlook uses "PST" files and Outlook Express uses "DBX" files. We might already suspect that the email was sent using Microsoft Outlook express, due to the email headers we received earlier.

    a) Search for files named *.pst or *.dbx by running "find -iname "*.dbx" -o -iname "*.pst"", you should be able to locate some outlook express databases in the "Test Users" folder

14) Now that we have located some emails files, we can use the undbx tool to extract the contents.

    a) Create a new folder to store the extracted emails, type "mkdir /cases/lab4/emails"

    b) Extract the individual emails from the .dbx files by typing "undbx -r /mnt/windows_mount/Documents\ and\ Settings/Test\ User/Local\ Settings/Application\ Data/Identities/\{178D21DB-5D77-4F2D-9803-518C1952042E\}/Microsoft/Outlook\ Express/ /cases/lab4/emails" (the TAB key can be a useful to help auto complete long file paths)

    c) Using the GUI navigate to the path /cases/lab4/emails in this folder you should locate additional folders for each of the emails extracted.

    d) Locate the emails in the sent item folder and double click on each one to open them

    e) When did Ima Hacker send the test message to himself?

    .....

    f) When did Ima Hacker send the message to Somepoor Victim?

    .....

    g) Notice that the email that Ima Hacker sent to Somepoor Victim has an email attachment, save a copy of the attachment to the email by right clicking the "client.zip.exe" file and selecting "Save As...". Save this in the /cases/lab4 directory.

15) Examine the email attachment

a) Make sure your terminal is in the /cases/lab4 directory.

b) Make sure the attachment was saved here by getting a directory listing (ls) the file should be named "client.zip.exe".

c) Scan the attachment for known viruses by running ClamAV. Execute the command "clamscan client.zip.exe" to scan only the saved email attachment (ignore the messages about ClamAV being outdated)

d) Note the output from ClamAV. Was there a virus detected? ……

e) Let's check the file size and hashes to see if it matches the sample given to us.
"ls -l client.zip.exe"

"md5sum client.zip.exe"

"sha1sum client.zip.exe"

compare the output to the details (file size, md5, sha1) given to you in the scenario section at the beginning.

Are they a match? ……

16) Let's also examine the email message in a bit more detail.

a) Type "head -n 20 /cases/lab4/emails/Sent\ Items/Ima\ Hacker_\ _imahacker72@yahoo.__somepoorvictim@yahoo.com.au__Illegal\ Website\ Access.000000000000F104.eml" to view the first 20 lines of the email.

b) Note the Microsoft Outlook Express version number and the exact date and time (including seconds) from within the email header:

Date and Time: .....

Outlook Express Version: .....

Check that these against the values in the email headers supplied to you.

Do they match? ……..

17) Looks like we've definitely found the email sent to Somepoor Victim!

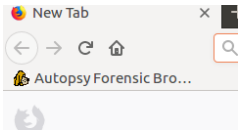
LAB 4: WINDOWS FORENSICS

4. EXAMINATION WITH AUTOPSY FORENSIC BROWSER

Autopsy is a graphical interface that makes using a lot of the sleuthkit tools (mmls, fls, icat, etc.) that we've been playing with a lot simpler.

CREATE AN AUTOPSY CASE

18) Open Autopsy Forensic Browser, and create a new case



a) Open the Firefox web browser then click on "Autopsy Forensic Bro…"

b) Create a "New Case", then enter "Lab4" in the "Case Name" field and click "New Case" again.

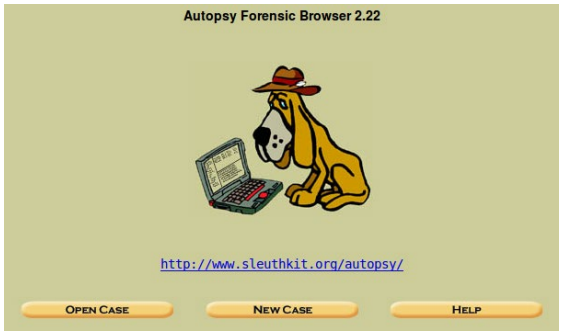


Figure 6. Autopsy Main Screen

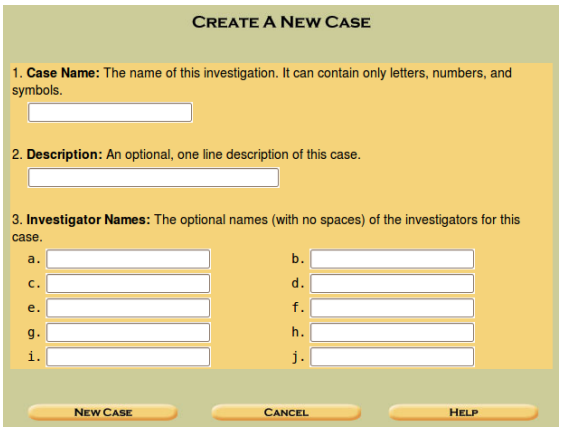


Figure 7. Autopsy "Create a New Case" Screen

c) Click "Add Host" to add details of the PC seized from Ima Hacker. On this screen, we can add details about the computer that the forensic image was taken from. We can also include details of any time skew if we were able to determine this by checking the PC's internal clock.
For this case, set the host name to "ImaHacker" and for the the Time zone to type exactly "Australia/ Melbourne" (without the quotes). Leave the other fields as is, and click "Add Host" to create the Host.

d) Our host now needs one or more forensic images associated with it. Click "Add Image", then "Add Image File" to add the image file we have in our /cases/lab4 directory. Under "Location", enter "/cases/lab4/ImaHackerRAT.E01". Type should remain as "Disk", and leave the import method as "Symlink". Click Next.

e) Autopsy automatically detects the partition type and sector offsets. Click "Add" to accept these details.

f) Click “ok” to go back to the “Host Manager” (if the computer had more than one hard disk drive, we could add additional forensic images by selecting “Add Image”)

g) In the “Host Manager”, select the “C:” volume, and click the analyse button.

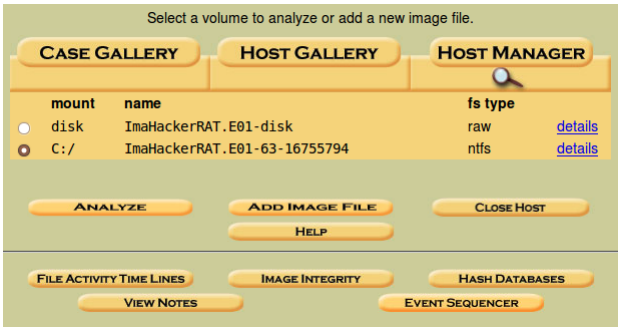


Figure 8. Autopsy Host Manager Screen Showing C: Volume for ImaHackerRAT.E01

h) Click “File Analysis” in the top menu bar. You’ll be presented with a list of files in the current directory (in this case, the disk opens at “C:\”, showing the “$” files at the top, then the standard windows files and directories (“Autoexec.bat”, “Documents and Settings”, “Windows”, etc.) below.

LOCATE MALWARE

19) Let’s have another go at locating the malware file named “client.zip.exe”

a) In the “File Name Search” box, type “client.zip.exe” and click “search”.

b) This time, we get a result – the file appears to sit under the “My Documents” folder. Click Show All Files and then Navigate to “Documents and Settings”, “Test User”, “My Documents”. You should now see a number of files that begin “Just a text file.txt:”, and a file named “Just a text file.txt”.

c) Go back to the terminal, and type
“ls "/mnt/windows_mount/Documents and Settings/Test User/My Documents"”
Compare this to the list of files shown in Autopsy.

There appear to be three more files visible in Autopsy that aren’t visible in a normal directory listing. These are called “Alternate Data Streams”. These are commonly used by windows to store metadata about files (such as if the file was downloaded from the internet, or an un-trusted network location). They can also be used by malware (or knowledgeable individuals) to hide files from view.

More Information (for later): http://en.wikipedia.org/wiki/Fork_%28file_system%29

d) Back in Autopsy, click on “Just a text file.txt:client.zip.exe” to view the contents of the alternate data stream (the “:” character means that there is an ADS of “Just a text file.txt” which is named “client.zip.exe”). You probably won’t be able to make out much, but observe the “File Type” details just above the contents of the file.

LAB 4: WINDOWS FORENSICS

Autopsy has automatically checked the file signature (remember we did this last week with "sorter"?), and has determined that the alternate data stream is, in fact, a Windows executable file ("PE32 executable for MS Windows GUI").

e) Click the "ASCII: report" button, and check the MD5 and SHA1 hashes against the hashes you were given earlier.

Do they match? ……..

f) Close the report tab, and click the "Add Note" button, then enter a comment (something like "Malware matches MD5 and SHA1 hashes provided"), and select all three dates to add this file to a list of important events. What are the times associated with this file?

……

g) Click "Event Sequencer" to see these events, then click "Close" to return to the host manager

TIMELINE ANALYSIS

Timeline analysis can be really helpful when we know roughly what time some events occurred, but we may not know exactly what happened or in what order.

20) Let's see what else was happening on the computer around the time of the executable file we just discovered

a) On the host manager screen, click the "File Activity Time Lines" button, then click the "Create Data File" button on the top of the window.

b) Select the "C:/" volume, and click the "ok" button. This first process collects all the times and dates from files on the volume.

c) Once the data file process is complete, click "ok" to create the timeline. Since this case is small, it won't take long to create the whole timeline. Leave the default values, and click "ok"

d) When the timeline's completed, click "ok". You'll be taken to the first event in the timeline (in 1999). Since we're interested in March 2010, enter "March 2010" in the date box, and click "ok".

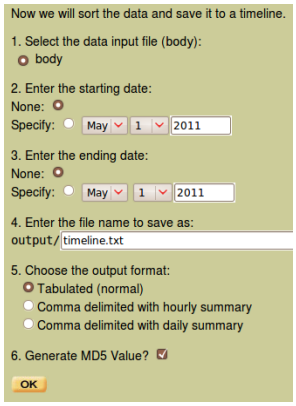


Figure9. Default Options for Timeline Creation

e) The columns shown by the timeline report are outlined below

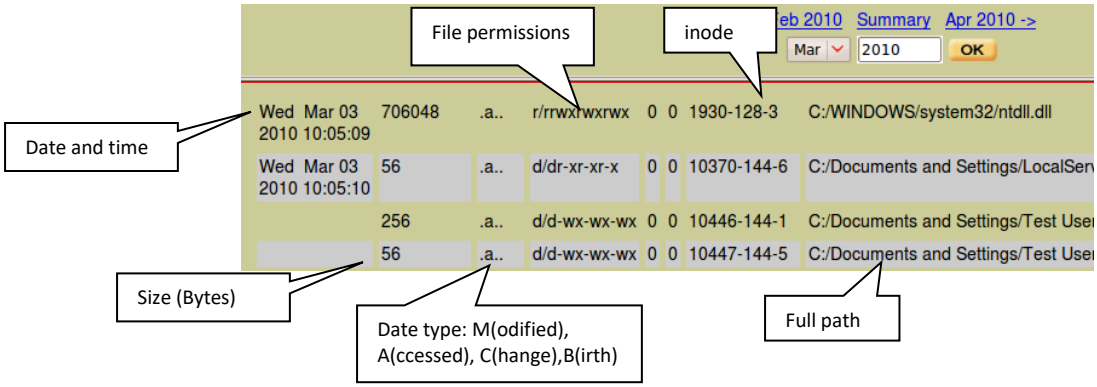


Figure 10. Screenshot Autopsy timeline window

f) Find the file entries beginning 10:10:24 on March 3rd. What type of user activity might these entries suggest?

.....

g) At 10:16:27, a file with the extension ".lnk" is created in the recent folder. Link files are created in the recent folder when a user opens a document from Windows Explorer (e.g. by double clicking an icon). Note the name of this file, as we'll come back to this later:

.....

h) Notice just prior to this link file (at 10:16:22 and 10:16:16), two cookie files are created. A cookie is a small bit of data that a website saves on your computer. This data can be used for tracking users or remembering users' website preferences. Note the names of these, as they could be related to our link file:

i) .....

ii) .....

i) At 10:17:23, a series of files and directories are created under the "My Documents" folder. What is the name of the main directory?

.....

j) There's one executable (.exe) file that gets created under this directory (at 10:17:24). What is its name?

.....

k) Also observe the "Zone.Identifier" alternate data streams (filename.ext:Zone.Identifier). These are created by windows when a file originates from a network location (such as the internet).

l) The existence of a file in the windows prefetch directory can tell us when a program was run. Use Firefox's "find" function (Press "CTRL-F", or select "Edit" then "Find") to locate references to the "client.exe" program. What times was it executed?

i) .....

ii) .....

✓ WINDOWS/Prefetch: Windows keeps track of which programs you commonly open to speed up opening applications. This information is stored as a number of small files in the prefetch folder. These files can often be useful in determining what programs have been run, and when.

m) There's some other activity that might be worth following up at 10:37:47. What do you think this might be?

.....

INTERNET HISTORY FILES

21) Since we found a lot of activity in the Temporary Internet Files folder, let's take a look at some internet history records to see what websites were visited.

a) Return to the "File Analysis Section" - first close the timeline view (click the 

button up the top right). Then select "C:/", click analyse and select the File Analysis view.

b) Conduct a "file name search" for "index.dat" (Internet Explorer stores its internet history records in files named index.dat).

c) Locate the file "C:/Documents and Settings/Test User/Local Settings/History/History.IE5/MSHist012010030320100304/index.dat" in the search results. Click on the file name in the top pane, and select the "ASCII Strings" display from the lower pane. You should see some URLs to google.com.au, some which include "*&q=...*". When a search is generated through Google, the URL includes the search terms entered by the user. The words after the "q" are these search terms. Note the Google search terms contained within this file:

i) .....

ii) .....

iii) .....

iv) .....

d) You should also notice a URL to a zip file from a website named "mediafire" (remember we also saw a cookie for the mediafire website earlier). What is the name of the zip file?

i) .....

22) Let's try decoding some of the information contained within the record for the zip file.

a) Go to the hex view for the index.dat file (click the "display" link next to the word "Hex"), and find the part of the file which refers to this zip file (*Hint: use Firefox's "find" feature, and search for ".zip"*). Notice that a little bit before the filename (offset 6280) are the letters "URL". This is the beginning of an Internet Explorer history record.

```
00006280:  5552 4C20 0200 0000 609E 0696 BABA CA01    URL ....`.......
00006290:  6026 9662 5EBA CA01 7C3C 0EBA 0000 0000    `&.b^...|<......
000062A0:  0000 0000 0000 0000 0000 0000 8051 0100    .............Q..
000062B0:  6000 0000 6800 0000 FE00 1010 0000 0000    `...h...........
000062C0:  0400 2000 0000 0000 0000 0000 0000 0000    .. ............
000062D0:  623C 0EBA 0100 0000 0000 0000 0000 0000    b<..............
000062E0:  0000 0000 0000 0000 3A32 3031 3030 3330    ........:2010030
000062F0:  3332 3031 3030 3330 343A 2054 6573 7420    320100304: Test
00006300:  5573 6572 4066 696C 653A 2F2F 2F43 3A2F    User@file:///C:/
00006310:  446F 6375 6D65 6E74 7325 3230 616E 6425    Documents%20and%
00006320:  3230 5365 7474 696E 6773 2F54 6573 7425    20Settings/Test%
00006330:  3230 5573 6572 2F44 6573 6B74 6F70 2F44    20User/Desktop/D
00006340:  6172 6B43 6F6D 6574 5241 5433 3346 5742    arkCometRAT33FWB
00006350:  2E7A 6970 0000 0000 0000 0000 0000 0000    .zip............
```

Figure 11. URL Record for DarkCometRAT zip File

b) The "Last Accessed" date for the URL record is stored at decimal offset 16 beginning from the "URL" header. The date itself is 8 bytes long. What is the hex value for the Last Accessed time of this zip file? Remember 2 hex characters represents 1 byte

.....

c) For now, let's just add a note to this file so we can come back to it later if required (click the "Add Note" button then enter a description. Create a sequencer event for all three times as well).

d) Click on "Event Sequencer" to see events you just created. Let's also create an event for the download of the zip file.

e) First, we now need to convert the hex value we noted before into a usable date and time. On your windows machine, visit http://www.digital-detective.co.uk/freetools/decode.asp, and download and run the DCode program. Save the file to your desktop (or somewhere easy to access), as you'll be using the program again later (the program doesn't need installation, so will run from anywhere).

f) Type the hex value you noted earlier into the "Value to Decode" box, leave the "Add Bias" at 00:00, and ensure the "Decode Format" box is set to "Windows: 64 bit Hex Value – Little Endian". Click "Decode".

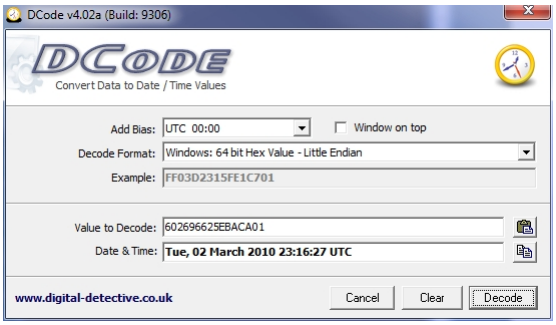


Figure12. Digital Detective DCode Application Window

g) Back in SIFT Workstation, add this date and time as a new sequencer event. Make the note something like "DarkComet Zip file downloaded". Set "Event Source" to "log", and click "Add"

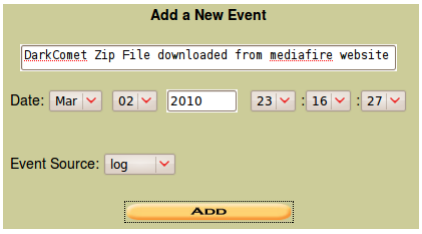


Figure 43. Add New Event to Autopsy Sequencer

LINK FILES

23) Remember the link file we located earlier? Let's go back and take a look at that.

a) Close the event sequencer, then in the host manager, select "C:/" then "Analyze" to continue analysing the forensic image.

b) Select File Analysis, and search for the link file you located earlier (hint: the file name for the link file was "DarkCometRAT33FWB.lnk"). View the contents of the file in the ASCII Strings display (click the filename in the top pane, then click "display" next to ASCII strings in the bottom pane).

c) Link files contain many helpful bits of information (dates and times, names and serial numbers of disk volumes, sometimes even the MAC address of the local computer), but for now we're just interested in the full path of the original file. Where was the DarkComet zip file originally located?

.....

d) Add a note to this link file (click "Add Note", type a suitable description, and add all three times to the sequencer). Click "ok" to save the note.

e) Click "View Notes" to see all the notes you've created so far. Then click "Close" to return to the Host Manager.

24) Let's take a look in the desktop directory to see if the zip file is still there.

a) Select the "C:/", and click "Analyze" then "File Analysis".

b) Navigate to "Documents and Settings", "Test User", "Desktop". It doesn't look like the DarkComet zip file is still there.

RECYCLE BIN / INFO2 RECORDS

25) Remember we also saw some activity involving the RECYCLER folder. This is the folder that Windows XP moves files to when they're deleted. Windows also stores information about the dates and times that files are moved there.

a) Navigate to the C:\RECYCLER folder (clicking the "C:/" up the top next to "Current Directory" will make it a little quicker!)

b) See how there's one folder (other than the "." and ".." folders – which mean "current" and "parent" directory) sitting in the RECYCLER folder? Each user has their own recycle bin, and the folder is named after the SID (Security Identifer) of the user.

The first part of the SID (up until the very last section) uniquely identifies the computer/operating system installation. The last four digits after the "-" uniquely identify the user account on that operating system.

What is the user part of the SID (i.e. the last four digits)?

.....

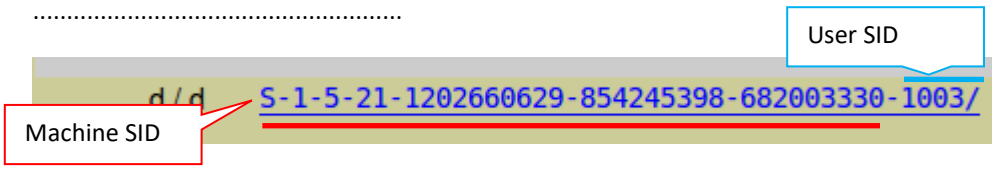


Figure 14. Recycle Bin Folder Showing SID Format

c) Open the user's recycle bin. You should see four files in this directory:

i) Dc1.bak – When windows XP moves a file to the recycle bin, it names it "D" for deleted, then the drive letter of the volume it was deleted from – e.g. a deleted file from "F:" becomes "Df" – then a number to identify the file. The file extension is left as is. We therefore know that this file used to be named *something*.bak, so it's not our zip file.

ii) desktop.ini (x2) – This is a file windows uses to tell itself that the folder is a special one. Desktop.ini files can specify folder icons or other settings for the folder.

iii) INFO2 – This is the file we're interested in. It contains the original path and deleted dates for all files contained within the recycle bin

d) Open the INFO2 file by clicking on the name. Click on the hex display. You'll see quite quickly that there's not a lot of content there. When a user empties the recycle bin, the file is reset to its default size of 20 bytes. Thankfully, we can often still recover some contents from the file in "slack space".

Since files are allocated space in a number of fixed-sized blocks or sectors, the actual "logical" contents (in this case, 20 bytes) can be considerably smaller than the total allocated space (the "physical" file).

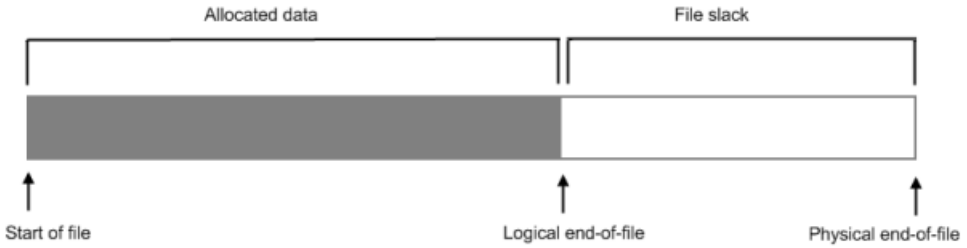


Figure 5. File Slack Diagram

e) Take a look at the more details about the INFO2 file by clicking the inode number for the INFO2 file (in the top pane, under the "Meta" column - the link named "12226-128-3").

f) If you scroll down to the bottom of this page, you'll see that there's a link to the disk block which holds this file ("16930") click this link to view the contents of this cluster (we could also have clicked the "Data Unit" button at the top of the screen, then entered the cluster number manually).

g) Click the "HEX" display button to view the contents of this cluster.

h) Scrolling down, the first string you'll see ("8C952042E}\Microsoft\OutlookExpress\Pop3ui.dl.bak") is the remaining part of the path associated with the first INFO2 record (the record for the deleted "Dc1.bak" file). Since a sector is the smallest bit of data that can be written to a file, when the INFO2 record was emptied, the first 512 bytes were overwritten with zeros (this is called "RAM slack", as the zeros are copied from RAM).

You may also note that each character in the string is taking up two bytes ("8.C.9.5.2.0.4.2.E."...). This is called Unicode, and is commonly used to store strings as it allows for non-English characters. ASCII (one byte per character) only has 256 possible characters, and thus isn't suited to many non-English languages.

i) Scroll down to offset 1616. There you should see the original path to our DarkComet.zip file.

j) If you scroll down a little further, you'll also see the path written in Unicode. At offset 1888 (just before the Unicode path) the deletion date and time is stored. What is the hex representation of this date? (8 bytes long)

.....

k) Back in Windows, put this date into DCode (same options as before – Bias = "00:00", Decode Format = "Windows: 64 bit Hex Value – Little Endian"). When was this file deleted?

.....