

SIRT Malware Toolkit Instructions

v. 1.0.3 – 8/1/2011

Introduction

The Oregon SIRT Malware Toolkit is an incident response tool to enable technicians with minimal forensic expertise to safely investigate a malware-infected system. Based on methodology developed by the Enterprise Security Office, users of the toolkit can identify malware components, investigate the origin, duration and vector of the malware infection, and use this information to assess the risk of sensitive data exposure.

Although the Malware Toolkit will maintain forensic integrity when used correctly, it is not meant for serious forensic investigations requiring rigorous methodology and practices. And, although the Toolkit enables simple malware analysis, it is not intended for in-depth malware reverse engineering. Full malware and forensic analysis both require extensive training to perform correctly and safely. They are both beyond the scope of this toolkit and methodology. The toolkit has been extensively tested with Windows XP systems but has been used with Windows 7 as well, although some functionality may not be available.

BE CAREFUL!

You will be handling live malware with the potential to infect your system and network. Although the Toolkit neutralizes a malware-infected machine by mounting the volume read-only using Linux, you, the analyst, must exercise caution too. Specifically:

- Isolate the Malware Toolkit system from your live network
- When handling files from the infected system, do not activate or open them with their native applications. For example, don't run executables, view images, read document files with their native applications on Windows. Don't use a web browser, even on Linux, to examine HTML, XML or Javascript files. Don't open and view PDF files on any platform using any PDF-viewing software. There are times when it is necessary to examine the contents of a potential malware file, e.g. , viewing a malware configuration file. If you need to examine the contents of a file, consider doing so from the Malware Linux environment, and use basic tools such as "less" and "strings" rather than full-featured document viewers.
- Ensure that your analysis machine is fully patched and updated.
- Ensure that your USB thumb drive has no autorun software, and disable autorun on your analysis machine.
- Even though the Toolkit and Methodology will identify malware on a system, it may fail to identify **all** malware. We recommend that infected systems be re-imaged before being returned to service.
- Clean up after yourself: delete malware samples from the analysis machine and/or thumb drive when finished with them.

You will need these things:

- SIRT Toolkit Boot CD
- Infected machine (powered off)
- Fully-patched analysis machine
 - May be Windows, Mac or Linux – but see below
 - Containing one or more high-grade anti-virus products, each different from the product on the infected machine
 - Anti-virus signatures must be up-to-date
 - Auto-cleanup or quarantine features must be disabled
 - Ideally with live secondary network connection (e.g. wireless)
- Crossover cable, switch or hub

- USB thumb drive with no autorun software and at least 200M free

Workflow

The analysis process consists of the following tasks:

1. Boot infected machine from toolkit CD and network it with analysis machine
2. If the system timezone is not Pacific (PST8PDT), set the system timezone
3. Plug the USB thumb drive in and begin system timeline generation
4. While timeline is processing, share drive and scan infected machine from analysis machine, locating malware files
5. Extract MBR; later you will scan or submit to virustotal.com
6. Submit malware files to virustotal.com to learn more about them
7. Determine infection time by locating all infected files on system timeline
8. Examine timeline and external logfiles and attempt to determine infection source

Instructions for each of these tasks are below.

Boot and Network configuration processes

1. Connect victim machine to analysis machine via crossover or hub/switch
WARNING: don't plug the victim machine into a live network. Although malware will not be active, the infected machine will be openly shared on the network
2. Boot victim machine from SIRT CD
WARNING: Ensure it boots from CD without bringing Windows OS up – booting Windows will reduce forensic integrity and reactivate malware
TIP: before booting, disconnect hard drive to set BIOS to boot from CD, then reconnect hard drive and boot from CD.
3. After the Malware Toolkit is finished booting, insert USB thumb drive into infected machine
4. Configure static IP network on victim machine
Click on System->System->YaST
Enter root password ("linux")
Click on "Network Settings" icon
Click Edit, Statically Assigned IP Address
set IP and netmask
Click Next, OK
5. Configure static IP network on analysis machine
6. Share drive from infected machine to analysis machine
Launch drive_share script from desktop
 - Click Run In Terminal
 - Select volume to share
 - Enter "y" to proceed
 - Wait
7. From Windows analysis machine, browse to and mount shared drive

Optional: Set System Timezone

The malware toolkit assumes that the system timezone is USA Pacific (GMT-7 or GMT-8, depending upon daylight savings time). If this is not the case, the system timezone can be set using the change_timezone tool.

1. Launch change_timezone from desktop
Click Run in Terminal
2. A menu of cities in United States timezones will be presented. They may scroll off the screen, use the slider bar to view entries at the top. Select the number corresponding to a city in your timezone. If you need an international timezone, select the last number in the

list to display world-wide timezones and select from them. Selecting the last number from the International timezones menu will return to the US listing.

Begin system timeline generation

Depending upon the size of the volume, the timeline generation process can take several hours to complete so it's important to get the process started as soon as possible.

Plug the USB thumb drive into the infected machine and:

1. Launch `timeline_generate` from desktop
Click Run in Terminal
If prompted, select volume from which to generate the timeline (your infected volume)
If prompted, select a device to write timeline to
(select your thumb drive, not `/media/root`, `/media/usb` or other system device)
Press ENTER to accept default filename for timeline file – this will prompt before overwriting a previous version. Alternately, enter a custom filename, beginning with `/` (it must be a full pathname and include your thumb drive device).
2. If registry hives are detected, the tool will prompt which hives to include in the timeline. If you know which user profile an infection may have occurred under, select that user's hive. Otherwise, select either "All" or "None" – looking at registry modification times in the timeline is rarely critical for determining source or time of a malware infection and increases the size and complexity of the resulting timeline file.
3. Note the system time on the infected machine. Is it accurate? The SIRT Toolkit creates all timeline info using Pacific timezone (unless set to something else as described above), but the system clock may be inaccurate. Look at the Toolkit clock in the upper-right corner and calculate the time offset between the infected machine time and real time – you'll need to know this to correlate timeline information with external information such as web content filter logs. Note that the time may be displayed in GMT because some systems use that internally but the SIRT Toolkit compensates for timezone differences – look primarily for inaccuracies in the minutes.
4. After the timeline has been generated you'll be asked whether you want to view it immediately. You may press "y" to view the timeline using the Unix "less" viewer (refer to Appendix C, below, for a "less" command reference). An alternative is to unmount the thumb drive, move it to a Windows machine, and import the timeline into Excel (although formatting will suffer) or use Wordpad (not Notepad) to view it.

Malware identification processes

- Once the two machines are networked and the timeline generation has been started, scan the mounted share volume from the analysis machine using one or more anti-virus products. Do a full scan, using all AV features. For best results, use an AV product that is different from what was running on the victim machine when it was infected.
- After scans have finished, submit located malware specimens to <http://virustotal.com> for multi-vendor identification information. This will also tell which anti-virus products have signatures for this malware. This can be done by either submitting the malware specimens directly from the Windows analysis machine or by copying them (using linux commands) onto the USB thumb drive, unmounting the drive (right-click -> unmount), then inserting it into a network-connected machine to submit. **Caution! Be careful not to activate (e.g. double-click on) malware specimens on your Windows analysis machine. Also do not open files with their native applications – e.g. web pages with a web browser.**
- Note the complete path and filename for each infected file. You will use this later to locate the infected files on the system timeline.

Extract Master Boot Record

Some malware (e.g. alureon, mebroot) installs itself in the MBR of a victim drive. You can extract a copy of the MBR and scan it for virus signatures.

Launch extract_MBR from Desktop
Click Run in Terminal

If prompted, select boot device from which to extract the MBR (your infected drive)

If prompted, select a device to write the MBR to

Press ENTER to accept default filename for the MBR file – this will prompt before overwriting a previous version – or enter a custom filename, beginning with “/” and including your thumb drive device path.

The extracted MBR file should be submitted to virustotal.com for analysis.

Timeline Analysis

Once the timeline generation process has finished you can start analyzing it. The Malware Toolkit generates several types of activity time data and integrates it into a timeline of activity on the system. The three main types of data are *file system* timeline data, *file metadata* information and *registry key modification* time data.

File System Timeline

File system timeline is generated from data associated with files on the system. Every file on the system has several dates associated with it that can be displayed by the timeline tool: the date the file was last modified, the date of the last time a file has been read or accessed, the date for the last time the MFT (Master File of Tables) entry for the file was modified, and the file creation date. These are, respectively, the “M,” “A,” “C” and “B” times, or “MACtimes.” The timeline tool reads these file dates and builds a report, sorted by date. Below is an altered example from an NTFS file system:

DATE / TIME	SIZE	MACB	PERMS	OWNER	GROUP	MFT	FILENAME
Tue Sep 22 2009 08:41:10	126	..b	r/rrwxrwxrwx	0	0	47198-128-1	/WINDOWS/file1
	56	m.c.	r/rrwxrwxrwx	0	0	48286-128-6	/WINDOWS/file2
	272	m.cb	r/rrwxrwxrwx	0	0	50902-128-1	/WINDOWS/file3
	280	..b	r/rrwxrwxrwx	0	0	50927-128-1	/WINDOWS/file4
	1180	.a..	r/rrwxrwxrwx	0	0	50132-128-1	/WINDOWS/file5
Tue Sep 22 2009 08:45:03	272	.a..	r/rrwxrwxrwx	0	0	50902-128-1	/WINDOWS/file3

For every file on the system there will be one to four entries: one for each of the M, A, C and B times. Changes to multiple file dates at once will be rolled into one entry, as illustrated above. Note that the “A” time for file3 is later than the combined “MCB” times.

File Metadata Timeline

File metadata includes date information found inside certain files on the system, including Windows event logs, Internet Explorer and Firefox history, system Restore Point and User Assist registry information, anti-virus logs, and other dated material. The Toolkit reads each of these files and integrates the dates into the timeline. Metadata timeline entries are indicated by square brackets “[]” around the source of the data in the Filename field – see Appendix B for an example including Internet Explorer browse history metadata.

Registry Key Modification Timeline

The modification dates for keys within individual registry hives can also be integrated into the timeline. Although this is useful for determining the registry-modification component of a malware attack, it isn’t essential for basic analysis of malware activity; it rarely helps determine when and how malware arrived on the system – except in the case of malware introduced via USB drive or file

share. The timeline generation tool allows selection of one or more (or none of) system and user hives for inclusion in the timeline. Registry Key timeline entries are indicated by the string "REG_" at the beginning of the Filename, followed by "System" or "User" followed by the name of the hive and the key path: e.g. "REG_System_system/ControlSet002/Control/IDConfigDB" is the system registry "system" and "REG_User_Administrator/Software/Microsoft/Windows/ShellNoRoam/Bags/6/Shell" is the Administrator user registry.

Find the earliest point of infection. Search the timeline file and locate each piece of malware identified by your malware scan, looking for the earliest occurrence of a malware file with an "m," "b" or "c" in the MACtime column. Also examine the timeline around times indicated by external data sources, such as an intrusion detection system alerts or firewall logs – although this type of data is often caused by ongoing activity rather than initial infection, locating these times on the timeline will give better understanding of the malware activity.

When looking for the first point of infection, look for combinations of timeline entries rather than a single isolated entry. Particularly useful are metadata entries such as Internet Explorer history. File system dates can sometimes be deceptive because they can be set and reset by various types of activity – for example, "m" file times can be inherited from the system a file originated on rather than being set when the file landed on the system you're investigating. Although not common, "m" file system times can also be reset by malware to deceive investigators. The best evidence of malware activity will be made up of multiple entries of various types at the same time.

Note that the ESO has frequently found pieces of malware much earlier than the current infection that turned out to be from previous undetected or partially neutralized infections. However, each of these previous infections were indicated on the timeline by multiple traces clumped together, not isolated files' timeline entries.

Locate additional malware. There will typically be several pieces of malware-associated activities clumped within a few seconds of each other, so look at the entries around the malware files you've already identified and locate other potentially-infected files that your anti-virus may not have found. Submit questionable files to virustotal.com to see if they might also be malware-related. Locate earliest timeline entries for each additional file you find to see if this changes your earliest point of infection.

Attempt to trace malware activity. Once the earliest point of the malware infection has been determined, examine timeline entries looking for clues to malware activity. Typically, malware infects a workstation in a series of stages, performing one exploit after another, fetching the next stage from the network when the previous stage is successful, until the workstation is fully infected. Using browser history and cache entries, try to identify successive stages, including malicious sites that provided them. Also look for indications of successful exploitation in the timeline such as files being placed in system directories, system reboots or event log errors.

Look at the minutes before and after the infection time for system activity such as software activation ("A" time entries on software files, Prefetch or User Assist entries on system software such as PDF viewers, Realplayer, Java, Flash, etc.) Although the "A" timeline records don't necessarily indicate infected files (e.g. system dll files), they can sometimes indicate what software was run during the infection process. If web-based exploit is a possibility, look at web history. The metadata timeline will include IE and Firefox history, cookie and cache file entries. Look for web requests to unusual sites that download javascript, executable files, or other unusual content, often followed by suspicious system activity. In association with browser history, look for exploit components in the browser cache. Another important thing to pay attention to is anti-virus activity during the suspected infection time – sometimes anti-virus software issues warnings or manages to neutralize part or all of an infection.

Don't forget to examine external data - network logs and even user reports can give indications of user and malware activity during the time of infection. These logs could include web activity logs from a proxy server or web content filter, email logs and firewall logs.

Identify actionable data surrounding the malware infection:

- Sites that can be blocked to prevent other infections
- Tell-tale indications of infection that can be used to locate other infected machines
- Successful exploitation of unpatched software that can be remediated to prevent other infections
- Unusual user activity that points to an educational opportunity
- Presence of data-stealing malware during periods when user has handled sensitive data – e.g. keystroke logger present when user logs into health information system

Optional – Add Registry Timeline Entries

Although examining registry key modification times isn't usually helpful to determine how a machine was infected from email or web browsing activities, it can sometimes be useful for identifying infections coming from other sources such as USB drives or file shares. In practice, we recommend not including registry information unless it's needed, and only necessary hives even then. The main problem is identifying whether a registry change resulted from malware or normal system activity. Nonetheless, if you wish to add registry modification information to the timeline, the malware toolkit gives you this capability.

In order to include registry information you need to go through the system timeline generation process as described above. When the selection of registry hives found on the system is presented (except the system "software" hive, which rarely adds much value but is huge), add the hives you're interested in one-by-one. The best strategy will usually be to only add the affected user registries (if you know them), and possibly the system "system" registry. When prompted whether to use existing system and metadata timeline data, answer "Existing" to include previously generated data rather than generating them from scratch. Generating registry timeline entries is relatively quick, typically taking only taking a few minutes.

Timeline Tips

- Read timeline entries in the context of surrounding activity. Isolated timeline entries that don't make sense are not good evidence of activity. Groups of entries are better evidence.
- "A" last-access dates are usually not relevant to determine malware infection times
- For File Activity timelines, "M" (file modified) dates are less reliable than "B" (created) dates (this doesn't count for Metadata or Registry timeline entries). M times can be inherited from the system the file was built on and are modifiable by the user or malware tool – although they sometimes accurately reflect when a file was created/modified, they sometimes are inaccurate.
- Remember that each type of timeline entry only shows the LAST time that type of activity happened – subsequent activity of the same type resets the timeline entry. For example, the MFT Modified date ("C") will only show the LAST time the MFT was modified; previous times will be lost.
- Registry modification dates sometimes get "wiped" by operating system patch activities. This will reset the date for the activity on all or most registers to the same date and time.

Appendix A – MACB Meaning by File System

Almost all malware cases will occur on Windows systems, so NTFS will be the file system to refer to below.

File System	m	a	c	b
Ext2/3	Modified	Accessed	Changed	N/A
FAT	Written	Accessed	N/A	Created
NTFS	File Modified	Accessed	MFT Modified	Created
UFS	Modified	Accessed	Changed	N/A

(source: The Sleuth Kit Mactime output: http://wiki.sleuthkit.org/index.php?title=Mactime_output)

Appendix B – Combined Metadata and System Timeline showing infection source and vector

The following timeline fragment is an extract of an actual malware case. Although this has been cleaned up to remove redundant entries, it shows evidence of the source of infection and vulnerability that was exploited to infect the machine. It has been displayed in table format to make it fit on the page and make it easier to read.

DATE / TIME	S I Z E	MACB	PERMS	O W N	G R P	MFT	FILENAME or [SOURCE INFORMATION]
Mon Dec 13 2010 09:26:15	0	macb	0	0	0	1882	[Internet Explorer] (Last Access) User: user123 URL:http://www.google.com/search?hl=en&source=hp&q=shortbread+cookie+recipe+paula+deen&aq=6&aql=g10&aql=&oq=shortbread+cookie&gs_rfai=CkQKaV1cGTewVCqW2iwOPp4FUAAAQgQFT9BEW1k cache stored in: /URL (file: /tmp-img/A/Documents and Settings/user123/Local Settings/History/History.IE5/index.dat)
Mon Dec 13 2010 09:26:27	0	m...	0	0	0	4049	[Internet Explorer] (Content viewed/Content saved to drive) URL:http://thiscrazybunch.com/svmgd/hfpdl.php?paula-deen-shortbread-cookie-recipe cache stored in: 9PZWY7MA/hfpdl[1].htm - HTTP/1.1 200 OK - X-Powered-By: PHP/5.2.14 - Keep-Alive: timeout=5- max=100 - Transfer-Encoding: chunked - Content-Type: text/html (file: /tmp-img/A/WINDOWS/Temporary Internet Files/Content.IE5/index.dat)
	0	m...	0	0	0	4049	[Internet Explorer] (Content viewed/Content saved to drive) URL:http://zonelink.co.cc/images/js.php cache stored in: YZMSDA0X/js[1].php - HTTP/1.1 200 OK - Content-Type: text/javascript - X-Powered-By: PHP/5.2.6-1+lenny9 - Content-Length: 1686 (file: /tmp-img/A/WINDOWS/Temporary Internet

	0	m...	0	0	0	4049	Files/Content.IE5/index.dat) [Internet Explorer] (Content viewed/Content saved to drive) URL:http://zonelink.co.cc/user/?name=d11&message=Flegg&e=snd cache stored in: SWXPM9WZ/index[1].html - HTTP/1.1 200 OK - Content-Type: text/html - X-Powered-By: PHP/5.2.6-1+lenny9 - Content-Disposition: inline; filename=index.html - Content-Length: 3923 (file: /tmp-img/A/WINDOWS/Temporary Internet Files/Content.IE5/index.dat)
Mon Dec 13 2010 09:26:28	6 1 1	.a..	r/rrwxr wxrwx	0	0	18875 -128- 5	/Documents and Settings/user123/Application Data/Sun/Java/Deployment/deployment.properties
	6 1 5 4 7	.a..	r/rrwxr wxrwx	0	0	19679 -128- 3	/Program Files/Java/jre1.5.0_06/bin/zip.dll
	3 2 8 7 8	.a..	r/rrwxr wxrwx	0	0	20908 -128- 3	/Program Files/Java/jre1.5.0_06/bin/hpi.dll
	1 1 8 9 0	.a..	r/rrwxr wxrwx	0	0	20914 -128- 3	/Program Files/Java/jre1.5.0_06/bin/java.dll

This shows the IE browse history of the victim. The victim followed a link to a Google search result for "paula deen shortbread cookie recipe," going to a legitimate site "thiscrazybunch.com." The site (or one of its advertising providers) had been compromised to link to "zonelink.co.cc" where malicious javascript was downloaded. Among other things, this invoked Java (note the "A" access entries for Java dll files). Not pictured is the download of a malicious jarfile that triggered a Java vulnerability.

Appendix C – Unix "less" command reference

The Toolkit's native file viewer is the Unix "less" command. Here are a few helpful commands for navigating in a file with "less." Note that all commands are case-sensitive.

Keystroke	Action	Description
h	help	help screen for a complete list of commands (press q to return to document)
q, Ctrl-C	quit	
G	Jump to end-of-file	
1G	Jump to beginning of file	
/[search term]	Search forward	Searches for term (regular expression) forward and sets search direction to "forwards"
?[search term]	Search backward	Searches for term (regular expression) backward and sets search direction to "backwards"
n	search again	Searches again for last search in current direction (forward or backward)
-l	toggles search case-sensitivity	Searches are case-sensitive by default
<space>, f	page down	Forward one page
b	page up	Backwards one page
<up arrow>, j	scroll up one line	
<down arrow>, k	scroll down one line	

<right arrow>	scroll right	Scroll right one half screen (if document is wider than the screen)
<left arrow>	scroll left	Scroll left one half screen (if document is wider than the screen)

Appendix D – Malware Report Template

The following report template (with examples) can be used to document the results of a malware analysis done using the Malware Toolkit. Customize this as necessary to fit your own needs.

Title: [something distinctive: e.g. "malware investigation of 12/15/10: Joe Smith workstation"]

Investigator(s):

Report Date: [12/15/2010]

Background

[describe trigger event, initial notification, symptoms, etc.]

Investigation Goals

[determine extent of infection, determine risk of data exposure, figure out how infected, etc.]

Key Questions and Answers:

- How did the malware infection occur?
[drive-by infection from site Y]
- When did the malware infection occur?
[Sept. 1, 2010 11:15AM]
- What vulnerabilities allowed the infection to occur?
[Unpatched Adobe Flash]
- What is the risk of data loss?
[High: Zeus on machine for 3 months]

Conclusions

[On Sept. 1, 2010, While browsing site Y in the normal course of business, Joe triggered a drive-by infection probably coming from a banner ad. The drive-by infection triggered a series of exploit steps, eventually resulting in installation of a trojan downloader and the Zeus trojan. Because Zeus is a data-stealing trojan, any sensitive information handled by Joe between Sept. 1st and the date of the investigation (December 1, 2010) should be considered potentially compromised.]

Evidence and Key data elements

[timeline entries showing evidence supporting conclusions, anti-virus or virustotal reports of malware types, etc.]

Followup Actions and Lessons Learned

[list actions that can/should be taken as a result of investigation plus recommendations]

1. [Joe should change passwords to all systems logged into during exposure window]
2. [recommend PII analysis of customer data during 3-month exposure windows to identify at-risk clients]
3. [accelerate application of Adobe Flash updates]

4. [Banner Ads should be blocked at web filter]