

Forum: [Hack Forums](#)

Thread: [Malware Analysis](#)

source - September 10, 2022, 8:22 pm

Forensic Cheat-sheets

<https://www.jaiminton.com/cheatsheet/DFIR/>

Online Services

[Virus Total](#)

[Hybrid Analysis](#)

[Koodous](#)

[Intezer](#)

Offline Antivirus and Detection Tools

Yara

Install

Code `sudo apt-get install -y yara`

Prepare rules

Use this script to download and merge all the yara malware rules from github:

<https://gist.github.com/andreafortuna/29c6ea48adf3d45a979a78763cdc7ce9>

Create the rules directory and execute it. This will create a file called `malware_rules.yar` which contains all the yara rules for malware.

Code `wget`

`https://gist.github.com/andreafortuna/29c6ea48adf3d45a979a78763cdc7ce9/raw/4ec711d37f1b428b63bed1f786b26a0654aa2f31/malware_yara_rules.py`

`mkdir rules`

`python malware_yara_rules.py`

Scan

Code `yara -w malware_rules.yar image #Scan 1 file`

`yara -w malware_rules.yar folder #Scan the whole folder`

YaraGen: Check for malware and Create rules

You can use the tool [YaraGen](#) to generate yara rules from a binary.

Check out these tutorials:

<https://www.nextron-systems.com/2015/02/16/write-simple-sound-yara-rules/>

<https://www.nextron-systems.com/2015/10/17/how-to-write-simple-but-sound-yara-rules-part-2/>

<https://www.nextron-systems.com/2016/04/15/how-to-write-simple-but-sound-yara-rules-part-3/>

ClamAV

Install

Code `sudo apt-get install -y clamav`
Scan

Code `sudo freshclam` #Update rules
`clamscan filepath` #Scan 1 file
`clamscan folderpath` #Scan the whole folder
IOCs

IOC means Indicator Of Compromise. An IOC is a set of conditions that identify some potentially unwanted software or confirmed malware. Blue Teams use this kind of definition to search for this kind of malicious files in their systems and networks.

To share these definitions is very useful as when malware is identified in a computer and an IOC for that malware is created, other Blue Teams can use it to identify the malware faster.

A tool to create or modify IOCs is [IOC Editor](#).

You can use tools such as [Redline](#) to search for defined IOCs in a device.

Loki

[Loki](#) is a scanner for Simple Indicators of Compromise.

Detection is based on four detection methods:

Code 1. File Name IOC

Regex match on full file path/name

2. Yara Rule Check

Yara signature matches on file data and process memory

3. Hash Check

Compares known malicious hashes (MD5, SHA1, SHA256) with scanned files

4. C2 Back Connect Check

Compares process connection endpoints with C2 IOCs (new since version v.10)

Linux Malware Detect

[Linux Malware Detect \(LMD\)](#) is a malware scanner for Linux released under the GNU GPLv2 license, that is designed around the threats faced in shared hosted environments. It uses threat data from network edge intrusion detection systems to extract malware that is actively being used in attacks and generates signatures for detection. In addition, threat data is also derived from user submissions with the LMD checkout feature and malware community resources.

rkhunter

Tools like [rkhunter](#) can be used to check the filesystem for possible rootkits and malware.

Code `sudo ./rkhunter --check -r / -l /tmp/rkhunter.log [--report-warnings-only] [--skip-keypress]`
PEpper

[PEpper](#) checks some basic stuff inside the executable (binary data, entropy, URLs and IPs, some yara rules).

NeoPI

[NeoPI](#) is a Python script that uses a variety of statistical methods to detect obfuscated and encrypted content within text/script files. The intended purpose of NeoPI is to aid in the detection of hidden web shell code.

php-malware-finder

[PHP-malware-finder](#) does its very best to detect obfuscated/dodgy code as well as files using PHP functions often used in malwares/webshells.

Apple Binary Signatures

When checking some malware sample you should always check the signature of the binary as the developer that signed it may be already related with malware.

Code #Get signer

```
codesign -vv -d /bin/ls 2>&1 | grep -E "Authority|TeamIdentifier"
```

#Check if the app's contents have been modified

```
codesign --verify --verbose /Applications/Safari.app
```

#Check if the signature is valid

```
spctl --assess --verbose /Applications/Safari.app
```

Detection Techniques

File Stacking

If you know that some folder containing the files of a web server was last updated on some date. Check the date all the files in the web server were created and modified and if any date is suspicious, check that file.

Baselines

If the files of a folder shouldn't have been modified, you can calculate the hash of the original files of the folder and compare them with the current ones. Anything modified will be suspicious.

Statistical Analysis

When the information is saved in logs you can check statistics like how many times each file of a web server was accessed as a web shell might be one of the most.

interbellum - September 11, 2022, 2:20 am

Really useful & good post bro. Keep up the good stuff :-)

Bastille - September 11, 2022, 3:29 am

Thanks for this thread OP, I'm currently installing FLARE-VM and didn't see it in your thread, is it deprecated ?

I'm following the playlist of Hackersploit on YT for malware analysis, 13Cubed also has a (short) playlist about it, but more focused on Forensics. Also bought "Practical Malware Analysis" book

which seems to be a good reference regarding this field.

Reaper34 - *September 11, 2022, 1:57 pm*

Awesome guide! I appreciate it!

ricardobecker123 - *September 20, 2022, 2:22 am*

Thanks for present these tools.