Malware analysis

Lecture 3

Basic Static Analysis

Basic Static Analysis

- Examining malware without executing it
- Quick and simple tools
- Reveal some immediate static information about the sample
- Ineffective against sophisticated malware

Cryptographic Hash Function

Definition

A hash function is a function that maps arbitrary long messages into a fixed length output.

H - hash function
m - message
H(m) - hash value

Examples:

- MD5 - 128 bit output
- SHA1 - 160 bit output
- SHA256 - 256 bit output

Cryptographic Hash Function

Properties

- Easy to compute

  Given m it is easy to compute $H(m)$
- Preimage resistance - one-way property

  Given $H(m)$ it is computationally infeasible to find a value m' such that $H(m') = H(m)$
- 2nd preimage resistance

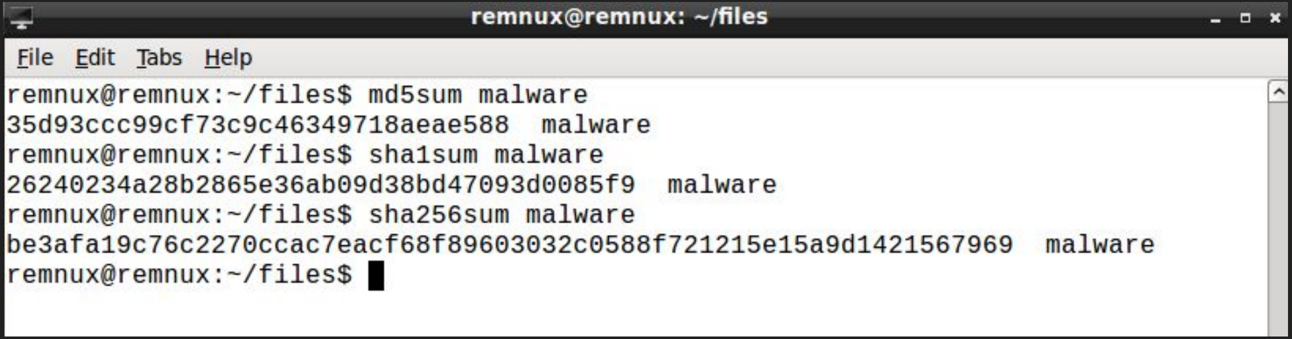
  Given m it is computationally infeasible to find a value $m' \neq m$ such that $H(m') = H(m)$
- Collision resistance

  It is computationally infeasible to find any m and m' such that $m' \neq m$ and $H(x) = H(x')$

Cryptographic Hash Function

In Malware Analysis

- Unique identifier for a malware sample
- Share with the community
- Use to find information about the malware



```
remnux@remnux:~/files$ md5sum malware
35d93ccc99cf73c9c46349718aeae588  malware
remnux@remnux:~/files$ sha1sum malware
26240234a28b2865e36ab09d38bd47093d0085f9  malware
remnux@remnux:~/files$ sha256sum malware
be3afa19c76c2270ccac7eacf68f89603032c0588f721215e15a9d1421567969  malware
remnux@remnux:~/files$ 
```

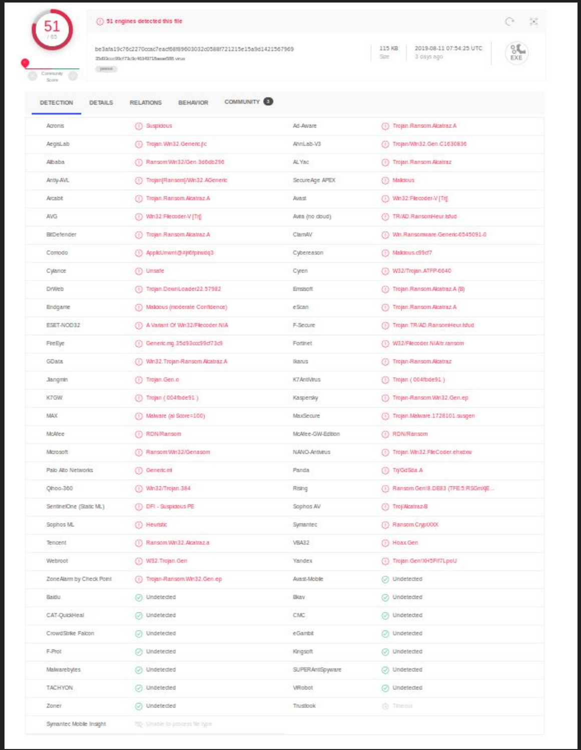
AV Scanning

www.virustotal.com

- Free service that analyzes files and URLs for malicious content
- Upload files to scan with multiple AV scanners
- Scan URLs
- Search for file hashes
- API available
- Uploaded files are available to anyone!!! -> search for hash

AV Scanning

- First step of malware analysis is to check if AV scanners detect the sample
- Signature names can help in identifying the malware family, but sometimes they are misleading or not informative at all
- Other valuable information
  - Basic static information
  - Behavior information
  - Submission information
  - Comments
  - Relations



51 / 65

Community Score

51 engines detected this file

be3afa19c76c2270ccac7eacf68f89603032c0588f721215e15a9d1421567969

35d93ccc99cf73c9c40349718aaae588.virus

peexe

115 KB
Size

2019-08-11 07:54:25 UTC
3 days ago

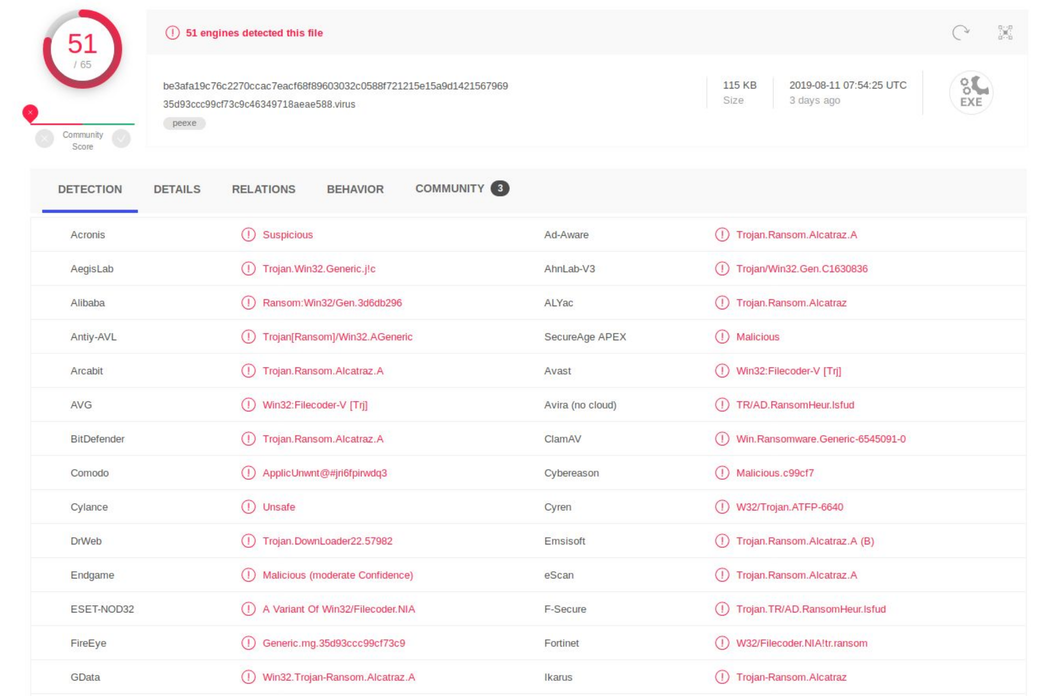
EXE

DETECTION | DETAILS | RELATIONS | BEHAVIOR | COMMUNITY 3

Acronis	Suspicious	Ad-Aware	Trojan.Ransom.Alcatraz.A
AegisLab	Trojan.Win32.Generic.j!c	AhnLab-V3	Trojan/Win32.Gen.C1630836
Alibaba	Ransom:Win32/Gen.3d6db296	ALYac	Trojan.Ransom.Alcatraz
Antiy-AVL	Trojan[Ransom]/Win32.AGeneric	SecureAge APEX	Malicious
Arcabit	Trojan.Ransom.Alcatraz.A	Avast	Win32:Filecoder-V [Trj]
AVG	Win32:Filecoder-V [Trj]	Avira (no cloud)	TR/AD.RansomHeur.lsfud
BitDefender	Trojan.Ransom.Alcatraz.A	ClamAV	Win.Ransomware.Generic-6545091-0
Comodo	ApplicUnwnt@#jri6fpiwdq3	Cybereason	Malicious.c99cf7
Cylance	Unsafe	Cyren	W32/Trojan.ATFP-6640
DrWeb	Trojan.DownLoader22.57982	Emsisoft	Trojan.Ransom.Alcatraz.A (B)
Endgame	Malicious (moderate Confidence)	eScan	Trojan.Ransom.Alcatraz.A
ESET-NOD32	A Variant Of Win32/Filecoder.NIA	F-Secure	Trojan.TR/AD.RansomHeur.lsfud
FireEye	Generic.mg.35d93ccc99cf73c9	Fortinet	W32/Filecoder.NIA!tr.ransom
GData	Win32.Trojan-Ransom.Alcatraz.A	Ikarus	Trojan-Ransom.Alcatraz
Jiangmin	Trojan.Gen.o	K7AntiVirus	Trojan (004fbde91)
K7GW	Trojan (004fbde91)	Kaspersky	Trojan-Ransom.Win32.Gen.ep
MAX	Malware (ai Score=100)	MaxSecure	Trojan.Malware.1728101.susgen
McAfee	RDN/Ransom	McAfee-GW-Edition	RDN/Ransom
Microsoft	Ransom:Win32/Genasom	NANO-Antivirus	Trojan.Win32.FileCoder.ehxdxw
Palo Alto Networks	Generic.ml	Panda	Trj/GdSda.A
Qihoo-360	Win32/Trojan.384	Rising	Ransom.Gen!8.DE83 (TFE:5:RSGmXjE...
SentinelOne (Static ML)	DFI - Suspicious PE	Sophos AV	Troj/Alcatraz-B
Sophos ML	Heuristic	Symantec	Ransom.CryptXXX
Tencent	Ransom.Win32.Alcatraz.a	VBA32	Hoax.Gen
Webroot	W32.Trojan.Gen	Yandex	Trojan.Gen!XH5Fif7LpoU
ZoneAlarm by Check Point	Trojan-Ransom.Win32.Gen.ep	Avast-Mobile	✓ Undetected
Baidu	✓ Undetected	Bkav	✓ Undetected
CAT-QuickHeal	✓ Undetected	CMC	✓ Undetected
CrowdStrike Falcon	✓ Undetected	eGambit	✓ Undetected
F-Prot	✓ Undetected	Kingsoft	✓ Undetected
Malwarebytes	✓ Undetected	SUPERAntiSpyware	✓ Undetected
TACHYON	✓ Undetected	ViRobot	✓ Undetected
Zoner	✓ Undetected	Trustlook	Timeout
Symantec Mobile Insight	Unable to process file type		

AV Scanning

- Some vendors are more reliable than others
- Different level of categorization
  - malicious (e.g. unsafe, generic)
  - malware type (e.g.ransom)
  - malware family (e.g.alcatraz)
- Always verify the results



51 engines detected this file

be3afa19c76c2270ccac7eacf68f89603032c0588f721215e15a9d1421567969
35d93ccc99cf73c9c46349718aeae588.virus
peexe

115 KB
Size

2019-08-11 07:54:25 UTC
3 days ago

Acronis	Suspicious	Ad-Aware	Trojan.Ransom.Alcatraz.A
AegisLab	Trojan.Win32.Generic.j!c	AhnLab-V3	Trojan/Win32.Gen.C1630836
Alibaba	Ransom:Win32/Gen.3d6db296	ALYac	Trojan.Ransom.Alcatraz
Antiy-AVL	Trojan[Ransom]/Win32.AGeneric	SecureAge APEX	Malicious
Arcabit	Trojan.Ransom.Alcatraz.A	Avast	Win32:Filecoder-V [Trj]
AVG	Win32:Filecoder-V [Trj]	Avira (no cloud)	TR/AD.RansomHeur.lsfud
BitDefender	Trojan.Ransom.Alcatraz.A	ClamAV	Win.Ransomware.Generic-6545091-0
Comodo	ApplicUnwnt@#jri6fpirwdq3	Cybereason	Malicious.c99cf7
Cylance	Unsafe	Cyren	W32/Trojan.ATFP-6640
DrWeb	Trojan.DownLoader22.57982	Emsisoft	Trojan.Ransom.Alcatraz.A (B)
Endgame	Malicious (moderate Confidence)	eScan	Trojan.Ransom.Alcatraz.A
ESET-NOD32	A Variant Of Win32/Filecoder.NIA	F-Secure	Trojan.TR/AD.RansomHeur.lsfud
FireEye	Generic.mg.35d93ccc99cf73c9	Fortinet	W32/Filecoder.NIA!tr.ransom
GData	Win32.Trojan-Ransom.Alcatraz.A	Ikarus	Trojan-Ransom.Alcatraz

https://www.virustotal.com/gui/file/be3afa19c76c2270ccac7eacf68f89603032c0588f721215e15a9d1421567969/detection

Malware Sample Sources

- https://www.hybrid-analysis.com/
- https://app.any.run/
- https://www.malware-traffic-analysis.net/
- https://beta.virusbay.io/
- https://thezoo.morirt.com/

Further sources: https://zeltser.com/malware-sample-sources/

Remember the rules that we discussed on the first lecture!!!!!

File Type

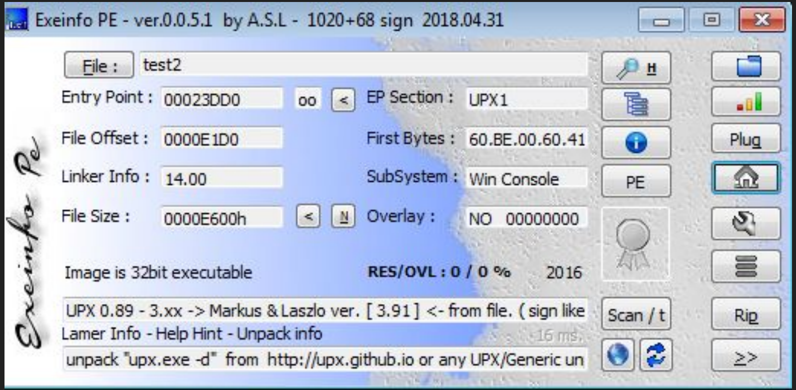
- Determine the file type
- Magic values e.g. PE files - MZ (4D 5A)
- Don’t trust extensions and icons
- file command on Linux
- https://filesignatures.net/

Strings

- Find strings (sequence of characters) in the analysed files
- https://docs.microsoft.com/en-us/sysinternals/downloads/strings
  - Scans the files for UNICODE or ASCII strings of a default length of 3 or more
- strings command on Linux
  - Default: at least 4 character long ASCII strings

Obfuscation

- Hide the purpose of the file. Make analysis and detection more difficult.
- String obfuscation: Try to hide the strings within the file, e.g. Base64, XOR, encryption.
- Packers: Obfuscate using compression, e.g. UPX
- Cryptors: Obfuscate using encryption.
- Tools: PEiD, Exeinfo PE, Notepad++



Exeinfo PE - ver.0.0.5.1 by A.S.L - 1020+68 sign 2018.04.31

File : test2

Entry Point : 00023DD0

EP Section : UPX1

File Offset : 0000E1D0

First Bytes : 60.BE.00.60.41

Linker Info : 14.00

SubSystem : Win Console

File Size : 0000E600h

Overlay : NO 00000000

Image is 32bit executable

RES/OVL : 0 / 0 % 2016

UPX 0.89 - 3.xx -> Markus & Laszlo ver. [3.91] <- from file. (sign like

Lamer Info - Help Hint - Unpack info 16 ms.

unpack "upx.exe -d" from http://upx.github.io or any UPX/Generic un

Questions

1. Upload the files to http://www.virustotal.com/ and view the reports. Does either file match any existing antivirus signatures?
2. When were these files compiled?
3. Where is the entrypoint of the file?
4. Are there any indications that either of these files is packed or obfuscated? If so, what are these indicators?
5. Do any imports hint at what this malware does? If so, which imports are they?
6. Are there any other files or host-based indicators that you could look for on infected systems?
7. What network-based indicators could be used to find this malware on infected machines?
8. What would you guess is the purpose of these files?
9. Use Resource Hacker to examine the resource, and then use it to extract the resource, where it is applicable. What can you learn from the resource?