# Malware Triage

Jordan Zeveney

Rubica, Inc.

RUBICA

# $ Whoami?

- 8 yrs in IT/IS
- Cyber Security Analyst, Rubica Inc.
  - IDS/IPS Triage, Network Forensics, Incident Response, & Malware Analysis
- CISSP, GCIH, GCFE, GCFA, Linux+
- Bachelors in Information Systems Security, AMU
- Working on Graduate Certificate in Incident Response,  STI
- Blog: https://ragingrooster.github.io/

# Overview

- This workshop serves as a rudimentary introduction to Malware Triage Analysis.
- Full scope Malware Analysis is often conducted over weeks, or months. Learn how to quickly extract indicators from a binary to determine if it is malicious, or not through this hands-on workshop.

# Workshop Requirements

- Laptop with at least 8GB of RAM
- VirtualBox: https://www.virtualbox.org/
- REMnux w/ VirtualBox Guest Additions installed: https://remnux.org/

# Workshop Outline

- Introduction ~5 mins
- What is Malware/Malware Analysis? ~5 mins
- Types of Analysis: Statics vs. Behavioral ~5 mins
- What is Malware Triage? ~10 mins
- Hands on Lab ~30 mins
- Q&A ~5 mins

# What is Malware?

mal·ware Dictionary result for malware

/ˈmalwer/

*noun*

software that is specifically designed to disrupt, damage, or gain unauthorized access to a computer system.

# Malware Types

- Adware – causes unsolicited pop-ups and advertisements to generate revenue.
- Backdoor – bypasses security controls and allows a remote attacker to execute commands on the system.
- Botnet – similar to backdoor, but consists of many compromised hosts issued the same command(s) at once.
- Downloader/Dropper – malicious code that downloads and installs additional malicious code.
- Hacktool – administrative tools, or programs that are abused by attackers.
- Hoax – delivers fake flags, or false warnings.
- Information-stealing – collects information and sends it back to the attacker. Includes sniffers, keyloggers, and hash grabbers.
- PUA/PUP – Potentially Unwanted Application, or Program installed without user’s knowledge, or approval. Often, adware related.

# Malware Types

- Ransomware – encrypts a user's device, or data and requests payment in exchange for the cryptographic key(s).
- Remote Access Trojan (RAT) – see Backdoor.
- Rootkit – malicious code that hides its existence.  User-level modifies, or replaces programs. Kernel-level manipulates the OS, creates backdoors.
- Scareware – scares a victim into purchasing something. Usually fake AV, or cleanup tools.
- Spam-sending – sends spam from an infected system.
- Trojan – malware that disguises itself as a legitimate program.
- Worm – malicious code that can replicate and infect other systems without user interaction.
- Virus – malicious code that can replicate, but requires user interaction.

Note: Malware doesn't typically fall neatly into just one of these buckets and can span multiple categories.

# What is Malware Analysis?

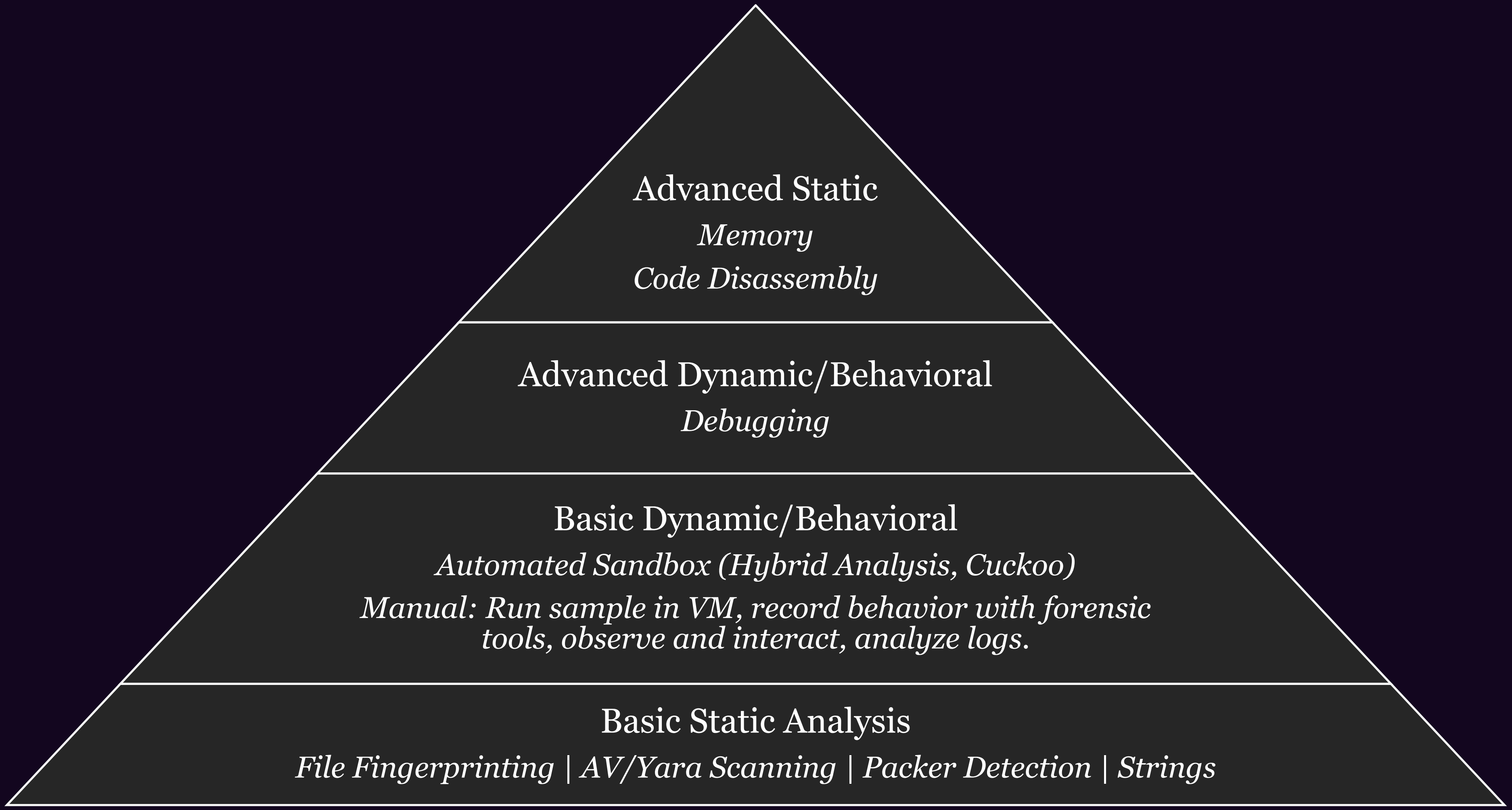
a·nal·y·sis Dictionary result for analysis

/əˈnaləsəs/

*noun*

1. detailed examination of the elements or structure of something.

# Types of Malware Analysis



Advanced Static

*Memory*

*Code Disassembly*

Advanced Dynamic/Behavioral

*Debugging*

Basic Dynamic/Behavioral

*Automated Sandbox (Hybrid Analysis, Cuckoo)*

*Manual: Run sample in VM, record behavior with forensic tools, observe and interact, analyze logs.*

Basic Static Analysis

*File Fingerprinting | AV/Yara Scanning | Packer Detection | Strings*

# What is Malware Triage?

- Reverse Engineering takes a lot of TIME (weeks, months, a year?)
- Sometimes we need information – fast!
- Triage involves gathering facts quickly and assigning a degree of urgency to a given sample.
- Advanced analyst/knowledge typically not required.
- This means Incident Handlers/Responders, Security Analysts, Forensic Investigators, and others can perform this type of analysis too.

# Triage – Golden Rules

- Define what you intend to gain from the analysis.
- Intelligence gathered during analysis should be actionable.
  - i.e. used to write Host-based, or Network Signatures
- Don’t get stuck in the weeds!
  - Malware can be complex.
  - Focus on key behaviors, or features.
- Don’t rely on one tool!
  - Remember there is no “one ring”, try multiple tools for integrity's sake.
- Everything *changes*.
  - Malware authors are smart. They change their tactics, techniques, and procedures (TTPs) to thwart analysis - analysts need to remain cognizant of this fact.

# Triage – What do we care about?

- Indicators of Compromise (IOC) – gather to mitigate, or hunt a sample.
- Examples:
  - IP Addresses
  - Domain Names
  - Autonomous System Names (ASNs)
  - Ports/Protocols
  - Hashes
  - Registry Keys
  - File Locations, Files Modified/Deleted/Created
  - Processes Stopped/Started
  - Mutexes
  - Strings
  - AV Signature
  - Yara Rule

# Triage – Questions to answer

- Where did the sample come from?
- What type of file is it?
- Has it been seen in the wild before?
- What is it capable of?
- What indicators can we extract?
- How can we safely remove it?

# Triage Checklist – Basic Static Analysis

| | Windows | Linux | MacOS |
|---|---|---|---|
| File Type: | > Get-FileType *filename* | $ file *filename* | $ file *filename* |
| File Magic: | HEX-Editor Plugin for Notepad++ | $ xxd *filename* \| head<br>$ hexdump –C –v *filename* \| head | $ otool –h *filename*<br>$ xxd *filename* \| head<br>$ hexdump –C –v *filename* \| head |
| File Hash: | > Get-FileHash *filename –Algorithm md5* | $ openssl dgst -md5 *filename* | $ md5 *filename* |
| Code Signature: | > sigcheck *filename* | *$ disitool.py extract *filename filename-sig.der*<br>*$ openssl pkcs7 –inform DER –print_certs –text –in *filename-sig.der > OUT_FILE*<br><br>Upload to VirusTotal | $ codesign –dvvv *filename* |
| ClamAV Detection: | > clamscan –ir filename | $ clamscan –ir *filename* | $ clamscan –ir *filename* |
| Yara Detection: | > yara *rulefile.yara filename* | $ yara *rulefile.yara filename* | $ yara *rulefile.yara filename* |
| Strings: | > strings *filename* | $ strings *filename* | $ strings *filename* |
| Resources (Imports/Exports/Libraries): | PEView; PE Explorer; CFF Explorer | $ pedump --imports *filename*<br>$ pedump –exports *filename* | $ otool –L *filename*<br>$ otool -l *filename* |

Note: This is a non-exhaustive list. There are a myriad of tools out there for malware analysis. Start with the OS your comfortable with then branch out.

# Triage Checklist – Basic Dynamic Analysis

| | Windows | Linux | MacOS |
|---|---|---|---|
| Network IOCs: | Wireshark; FakeDNS; INetSim | Wireshark; TCPDump; ngrep; FakeDNS; INetSim | Wireshark; FireEye Monitor.app;<br>$ lsof<br>$ netstat |
| Registry: | Regshot; Noriben.py; VirusTotal Sandbox, Hybrid Analysis, Cuckoo | VirusTotal Sandbox, Hybrid Analysis, Cuckoo | VirusTotal Sandbox, Hybrid Analysis, Cuckoo |
| Running Processes & Services: | Noriben.py; Process Explorer; Process Hacker; Process Monitor; VirusTotal Sandbox, Hybrid Analysis, Cuckoo | $ top<br>$ netstat<br>$ lsof -i<br>$ lsof –c <SERVICE NAME><br>$ lsof –p <PID><br>$ ps –aux | Activity Monitor; FireEye Monitor.app;<br>$ top<br>$ netstat<br>$ lsof -i<br>$ lsof –c <SERVICE NAME><br>$ lsof –p <PID><br>$ ps |
| Files Created/Modified/Deleted: | Noriben.py; VirusTotal Sandbox, Hybrid Analysis, Cuckoo | VirusTotal Sandbox, Hybrid Analysis, Cuckoo | FireEye Monitor.app; VirusTotal Sandbox, Hybrid Analysis, Cuckoo |

Note: This is a non-exhaustive list. There are a myriad of tools out there for malware analysis. Start with the OS your comfortable with then branch out.

# Lab Time

- 30 mins
- Use REMnux to analyze a suspected maldoc
- Download the sample here <>

# Lab Walkthrough

- Determine the file type:
  - $ file Evil.docm
  - $ xxd Evil.docm | head
- Generate Hashes:
  - $ openssl dgst -md5 Evil.docm
  - $ openssl dgst -sha1 Evil.docm
  - $ openssl dgst -sha256 Evil.docm
  - $ ssdeep Evil.docm

# Lab Walkthrough

- Research Hashes:

```
$ python Automater.py a102976763e24de9871be806a0f18ba1
$ python Automater.py 40adac8fe197a9c3cf3ab965ad897cfd45e14c4e
```

- Scan with AV:

```
$ freshclam
$ clamscan -ir Evil.docm
```

- Unpack AV Signatures:

```
$ locate daily.cvd
$ cp /var/lib/clamav/daily.cvd ~/Downloads/
$ sigtool -u daily.cvd
$ locate main.cvd
$ cp /var/lib/clamav/main.cvd ~/Downloads/
$ sigtool -u main.cvd
```

# Lab Walkthrough

- Determine what’s in the AV signature:

  ```
  $ grep 'Doc.Downloader.Generic-6680573-0' *
  $ echo '4174747269627574652056425f4e616d65203d2022' | xxd -r -p
  $ echo '22706f7765727368656c6c2e657865' | xxd -r -p
  $ echo '28286e65772d6f626a656374' | xxd -r -p
  $ echo '286578656329' | xxd -r -p
  ```

# Lab Walkthrough

- Scan with Yara:
  ```
  $ yara -gms /opt/remnux-rules/yara/Malicious_Documents/Maldoc_VBA_macro_code.yar Evil.docm
  ```
- Look for strings:
  ```
  $ strings Evil.docm
  ```
- Find the macro:
  ```
  $ python /opt/remnux-scripts/officeparser.py Evil.docm
  ```
  ```
  $ python /opt/remnux-didier/oledump.py Evil.docm
  ```

# Lab Walkthrough

- Dump the macro:
  - $ oledump.py -s 3 Evil.docm
  - $ oledump.py -s 4 Evil.docm
- Dump the macro and translate to readable format:
  - $ oledump.py -s 3 -v Evil.docm
  - $ oledump.py -s 4 -v Evil.docm
- Submit the sample for automated behavioral analysis:
  - Virus Total
  - Hybrid Analysis

# Q&A

~ 5 mins

# References & Resources

- Incident Response & Computer Forensics, 3rd Edition by Kevin Mandia, Matthew Pepe, Jason Luttgens. Publisher: McGraw-Hill Osborne Media. Release Date: August 2014. ISBN: 9780071798693.
- OSX Incident Response Scripting and Analysis by Jaron Bradley. 2016 Elsevier Inc. ISBN: 9780128045039.
- Practical Malware Analysis by Andrew Honig, Michael Sikorski. Publisher: No Starch Press. Release Date: February 2012. ISBN: 9781593272906.
- Malware Analyst's Cookbook and DVD: Tools and Techniques for Fighting Malicious Code by Matthew Richard, Blake Hartstein, Steven Adair, Michael Hale Ligh. Publisher: John Wiley & Sons. Release Date: November 2010. ISBN: 9780470613030.
- Advanced Malware Analysis by Christopher Elisan. Publisher: McGraw-Hill. Release Date: September 2015. ISBN: 9780071819756.
- Malware: Fighting Malicious Code by Lenny Zeltser, Ed Skoudis. Publisher: Prentice Hall. Release Date: November 2003. ISBN: 0131014056.
- Learning Malware Analysis by Monnappa K A. Publisher: Packt Publishing. Release Date: June 2018. ISBN: 9781788392501
- Blue Team Field Manual (BTFM) by Alan White & Ben Clark. 2017. ISBN: 154101636X.
- https://ragingrooster.github.io/