CMSC 426
Principles of Computer Security

Lecture 09

Malware Analysis

Last Class We Covered

- Malware categories
  - How it spreads
  - What it does
  - What kinds of systems it targets

- Malware lifecycles

Any Questions from Last Time?

Today's Topics

- Indicators of Compromise
  - Hashing

- Analysis
  - Basic/Advanced
  - Static/Dynamic
  - Packers and Sandboxing

- Info on Exam 1

Indicators of Compromise

Review: Indicators of Compromise

- Evidence that malware was on a system/network
- Can be used for attribution to a malware family, actor, and/or campaign

- Examples:
  - IP addresses and domain names
  - Email addresses
  - Cryptocurrency wallets
  - Hashes

IP Addresses and Domain Names

- Can show up in different instances:
  - IP address or domain name the malware downloaded from
  - IP address or domain name that the malware uses for C&C

- Quick reminder:
  - IP address:
    - 192.168.0.1
  - Domain name:
    - google.com

Email Addresses

- Can show up in different instances:
  - Email address used to send a phishing email
    - (May be spoofed, however)
  - Email address used to register a domain name
    - Not actually provided in the malware, but possible to look up who registered the domain name
    - With that information, possible to find out what other domains have been registered by that actor

Cryptocurrency Wallets

- Can show up in different instances:
  - Wallet listed in a ransomware note
    - Easy to find, for obvious reasons
  - Wallet that a cryptocurrency miner “deposits” into

Hashes

- Unique large number calculated by a hashing algorithm
    - In other words, the output of the hashing algorithm
    - Sometimes called the “digest,” often just called the “hash”

- If two files share the same hash, there is an *exceedingly* high probability that the files are identical

- Hashing algorithm may be run on any malware file
    - Files in payload, in first stage, etc.

(Not Your) Data Structures Hash

- What's the goal of hashing in data structures?
  - Placing data of a larger domain into a table of a smaller domain
    - Quick insertion, traversal, and retrieval are key
    - Need to minimize collisions at various hash table sizes
    - Fast performance of hashing algorithm (for resizing)

- In this context, that is <u>not at all</u> what the focus is
  - Speed of hash calculation is only vaguely important
  - Will not mod the hash result, so collision avoidance is easier

Hash Digest Similarities

- If two files have the same hash, they are functionally identical
  - Sort of allows a “diff” without having both files together

- If even one small change is made, the hash will change *drastically* (may be entirely different)

- Different hashing algorithms generate different sizes of hash
  - MD5, SHA1, and SHA256 are most common algorithms
  - (16, 20, and 32 byte hashes are generated, respectively)

Import Table Hashing

- Import address table is metadata within payload files
  - Contains list of all library functions used, in order they appear in code
  - Created by the original compiler/linker as the file is compiled/linked

- Hashing the import table gives you an imphash
  - “import hash”

- If hashing the whole file, a single change → different hash
  - If using an imphash, changes would have to be more substantial
  - But still unique-ish – variants will likely have different import tables

Fuzzy Hashing

- Official name is “context triggered piecewise hashing”
  - Most common program used for this is called ssdeep

- Details of how it works are complex, but essentially:
  - More robust against changes than traditional hashing
  - Can compare two fuzzy hashes and get a similarity score

Malware Analysis

Malware Signatures vs Behavior

- Two different aspects of malware that can be analyzed

- Signature
  - Aspects of the malware that show up “at rest”
  - Strings and byte sequences
- Behavior
  - Actions the malware takes when run
  - API functions called, etc.

Basic Static Analysis

- Examining the malware while it is “at rest”

- Plain-text strings within the code
- Hashes (MD5, SHA-1, imphash, fuzzy)
- Functions used (Windows API, etc.)
- General information (malware type and family)
- Other known instances of the malware

Basic Dynamic Analysis

- Observing the output and/or changes when the malware is run
  - But not interfering or interacting with the malware

- Debug/error messages the malware outputs
- Changes to the registry

Advanced Static Analysis

- Examining the malware's code (assembly) in detail
- Disassemblers organize the code into subroutines, and allow the analyst to more easily trace their way through the code
  - Much, much easier than reading the raw assembly
- This information is normally used to inform what actions the analyst takes in the debugger

Advanced Dynamic Analysis

- Using a debugger to control any and all aspects of the malware as it is being executed
  - Registers, stack, memory, and code

- In the demo, we will see this used to “trick” the malware into accepting any *incorrect* password as correct

	Static	**Dynamic**
Basic	Looking at details of the malware when it is “at rest” *ex*: virusTotal, strings	Running the malware and observing changes/output *ex*: regShot, DebugView
Advanced	Closely examining the malware’s code in detail *ex*: IDA Pro	Running the malware and using a debugger to control details of its execution *ex*: ollyDbg

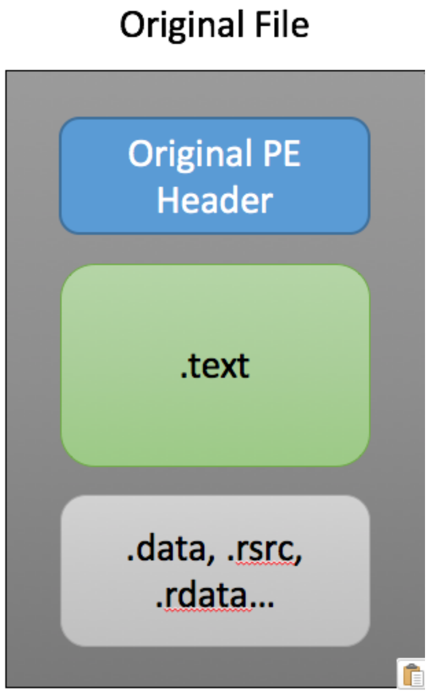
More Malware Analysis Info

Malware Packers

- Goal is to obfuscate information about the malware
  - Code, strings, and sometime imports
  - Makes the malware more difficult to analyze
- Does this by compressing and/or encrypting the malware
  - Simpler for the attackers than directly implementing protection within the code itself
- Decrease chance of detection and increase amount of time/effort required for effective analysis

Information taken from https://securingtomorrow.mcafee.com/business/malware-packers-use-tricks-avoid-analysis-detection/

Malware Packer Example



Information taken from https://securingtomorrow.mcafee.com/business/malware-packers-use-tricks-avoid-analysis-detection/

Sandboxing

- Automated technology for malware detection
  - Sandbox attempts to analyze the malware automatically

- Place malware into a closed, controlled environment
  - Simpler setup; less complex environment

- Reasons for using sandbox
  - Can’t cause any lasting damage
  - Easier to analyze

Information taken from https://www.apriorit.com/dev-blog/545-sandbox-evading-malware

Sandbox Evading

- Malware can attempt to recognize if it’s in a sandbox
  - Won’t do anything malicious if it realizes this is the case

- Some techniques include:
  - Not running unless certain dll files are available (many of which are not included in the sandbox)
  - Running at a specific date/time
  - Requiring user interaction (sandbox is automated)

Information taken from https://www.apriorit.com/dev-blog/545-sandbox-evading-malware

Announcements

- Homework 1 will go up on the course Blackboard
  - Due at midnight on Wednesday, March 13th
  - Essentially an exam review sheet
- Lab 2 will come out later this week
- Midterm 1 is on Thursday, March 14th

Midterm Info and Review

Exam Rules

- The midterm is closed everything:
  - No books
  - No notes
  - No cheat sheets
  - No laptops
  - No calculators
  - No phones

Exam Rules

- Place your bag under your desk/chair
  - NOT on the seat next to you
- You may have on your desk:
  - Pencils, erasers
    - You **must** use a pencil, not a pen
  - Water bottle
  - **UMBC ID**
    - You **must** bring your UMBC ID with you to the exam! We won't accept your test without it.

Exam Rules

- Your TA or instructor may ask you to move at any time during the test
  - This doesn’t mean we think you’re cheating

- That being said, **DO NOT CHEAT!!!**
- Cheating will be dealt with severely and immediately
  - There will be no retakes or partial credits

Exam Format

- Multiple Choice
- True/False
- Short answer
  - Similar difficulty to questions on homeworks/labs

Exam Content

- Heavy on stack overflow attacks, medium-light on malware

- Very little you should need to memorize by rote
  - Not going to ask about many specific pieces of malware
  - Very few acronyms will be used

- Exam is designed to test actual knowledge and understanding
  - If you don’t manage to complete Lab 1, talk to someone who did (or come to office hours)

Exam Advice

- When you first get the exam...

- Write down your name
  - Make sure your name is ***legible*** and on the line
- Circle your section number
- Read the Academic Integrity agreement
  - Sign your name underneath

Image Sources

- Bitcoin wallet (adapted from):
  - https://www.flickr.com/photos/30478819@N08/24874103608