CMSC 426
Principles of Computer Security

Lecture 07

Introduction to Malware

Last Class We Covered

- Defenses against stack overflow attacks
  - ASLR
  - Stack canaries
  - Preventing stack execution

- Buffer overflow variations
  - return-to-libc
  - Return-oriented programming

Any Questions from Last Time?

Today's Topics

- Malware
- Threat actors
  - APT groups and others

- Attribution
- Threat actor examples

- Malware categories
  - How it spreads

Why Hack Systems?

- Stack overflow attacks can let us gain control of a system (among other things), but what do you do with them?

- What is the end goal?
  - Notoriety
  - Money
  - LOTS of money
  - Political influence

Malware

- Short for “malicious software”
  - May attack applications, editors/compilers, or the kernel level
  - Often delivered through compromised websites or spam emails

- May be silent, logging keystrokes (*e.g.*, passwords)
- May be annoying, constantly popping up advertisements
- May be disruptive, disallowing use of certain programs or parts of the system
- May be exploitative, using cycles or sending mass emails

Threat Actors

Three Classifications

- “Script kiddies”
  - Basic, low stakes
- APT groups
  - Well funded, world players
- Cybercriminals
  - Generic threat actor

"Script Kiddies"

- Largely unskilled
  - Not necessarily young, despite the name
- Use scripts and code created by others

- Often vandalize websites or attack systems and networks
  - Sometimes assumed to not know/understand the consequences

- End goal is street cred, sense of superiority, petty crime

APT Groups

- Advanced
  - Use a wide variety of tactics (including custom malware) specifically chosen for the target
- Persistent
  - Attacks may happen over an extended period against a chosen target, maximizing the chance of success
- Threat
  - Focus on a specific target by experienced, well-funded attackers
  - Often actively involved, instead of simply using automated tools

More on APT Groups

- APT groups are often funded by a specific country
- Countries do not normally admit to this
  - Makes more sense to keep details and information secret

- End goal varies based on the target
  - Information, influence, money, large amounts of personal data

Cybercriminals

- Higher skill level than script kiddies, less organized and well-funded than APT groups
  - Essentially anything that's not the other two
- May work alone or in groups
- End goal is generally money
  - Either directly (scams, hacking financial institutions) or indirectly (planting ransomware, selling access to created botnets)

Threat Actor Examples

APT 1 ("Comment Crew")

- Exposed in 2013 as being formed of a military group from China
  - People's Liberation Army Unit 61398
- Has stolen massive amounts of data from organizations
  - Hundreds of terabytes
  - Data includes blueprints, proprietary processes, and contact lists
  - Focus on English-speaking countries
- Maintain access to systems for nearly a year on average, continually revisiting and stealing additional data

Information taken from https://www.fireeye.com/current-threats/apt-groups.html

APT 28 ("Fancy Bear")

- Likely associated with the Russian government
  - US Special Counsel believes it is two GRU units
  - GRU is Russia's military intelligence agency

- Partially responsible for the DNC hack in early 2016
- Also attacked 2017 elections in France and Germany

- Main goal seems to be political influence

Information taken from https://www.fireeye.com/current-threats/apt-groups.html and https://en.wikipedia.org/wiki/Fancy_Bear

Attempting Attribution

TTP (Tactics, Techniques, and Procedures)

- Analyzing the information on how attacks are managed and accomplished to try to identify the group(s) responsible

- Some examples of TTPs:
  - What were the tactics/techniques are used in the attack?
  - How was information gathered prior to attack being carried out?
  - How was the payload delivered?
  - What was the timeline for the attack?
  - What was the target's type?

IOC (Indicators of Compromise)

- Using evidence left behind to identify the group(s) responsible

- Examples:
  - Exact malware used by group (might have been seen before)
  - Infrastructure used for attack
    - IP addresses, domain names, etc.
  - URLs/domain names of botnet Command & Control servers
  - Bitcoin wallet
  - Metadata about the above

Difficulties of Attribution

- Even with this information, can be difficult to attribute attacks
  - Evidence is often ambiguous, or even contradictory
  - Who the target is can also be a factor in attribution

- Possibility of false attribution can also be a problem
- Some groups deliberately leave “fingerprints” after their attack
  - These fingerprints may be deliberate false flags

Malware

Categorizations

- Malware is categorized based on three factors
  - How it spreads/persists
  - What it does
  - What kinds of systems it targets

- A single piece of malware can belong to more than one classification within a category
  - Classifications are fuzzy and overlap
  - These are just general guidelines, not a taxonomy

How Malware Spreads

Worm

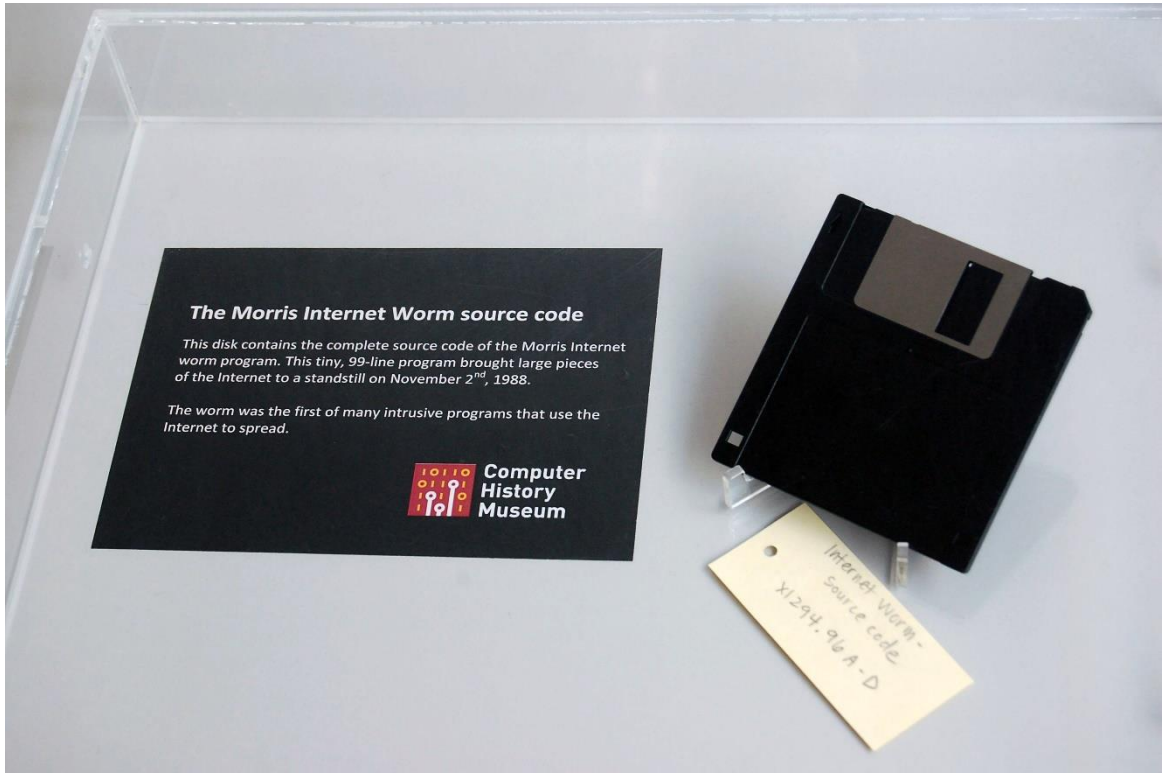
- Standalone program
- Replicates itself and spreads automatically
  - Attempt to infect as many computers as possible
- Normally spread via a network
  - Consumes bandwidth; dangerous even if “harmless”
- Usually exploits a vulnerability to do so
  - Or uses previously captured authorization credentials

Worm Example: Conficker

- Exploits the MS08-067 vulnerability (an overflow vulnerability!)
  - Vulnerability was patched before the worm came out
- Still propagating a decade later
  - Mostly on unpatched legacy systems
- Estimated 9 to 15 million computers infected since 2008
- The authors of the worm still have not been identified

Worm Example: Morris Worm

- Released by grad student Robert Morris in November 1988
  - Claimed it was meant to gauge the size of the Internet
  - Debate over his true intentions



- Infected about 10% of computers connected to the Internet in 1988
- Spreading mechanism led it to re-infect machines, which slowed or crashed them

Worm Example: Morris Worm (cont)

- Once it was on a system, it obtained a list of all known hosts that would allow entry from the current host
- Then tried to gain access to each one, by either
  1. Attempting to log on as a legitimate user, using a simplified brute force method of password cracking
  2. Exploit a bug in the **`finger`** protocol
  3. Exploit the debug option of the mail receiving program
- Infected systems would respond they were infected
  - 1 out of 7 times, the worm would propagate regardless

File Infector

- Also commonly called a virus
  - (But ***not everything*** is a virus! Watch your language!)

- Inserts its own code into executable files to persist and spread
  - Code is now “infected code”
  - When the infected executable is run, the virus also executes

- Virus is spread when the infected executable is copied onto another system or otherwise spread

Trojan (or Trojan Horse)

- Malicious program that appears to have a useful function
- Often spread by social engineering
  - Executing email attachments
  - Clicking on advertisements
- Payloads can be a variety of things, including backdoors, ransomware, etc.

Daily Security Tidbit

- September 2018, the creators of the Mirai botnet were sentenced to probation (instead of jail time)
  - Provided “extraordinary cooperation” with the government
- Mirai infects and takes over things like routers and DVRs
  - Then uses them in large-scale botnet attacks like DDoS
  - Creators rented out “slices” of the botnet to other cybercriminals
- Released the code in an attempt to obscure their authorship
  - Copied by others, and used to cause even more damage

Information taken from https://krebsonsecurity.com/2018/09/mirai-botnet-authors-avoid-jail-time/