Malware Mitigation

Chengyu Song

Slides modified from
Heng Yin, Vern Paxson and Dawn Song

Lab1: reverse engineering

- Goal: understand what the program does and how it works
- Approaches
  - Static: disassembler (objdump, radare2, IDA)
  - Dynamic: debugging (gdb)
- Why useful?
  - QA: make sure the code is correct
  - Bug fixing: figure out why
  - Malware analysis

Malware detection

- Static signature based approach
  - Countermeasures from malware authors
- Dynamic behavior based approach
  - Countermeasures from malware authors
- Network based approach
  - Worm detection and botnet take down

Malware analysis

- To answer following questions
  - Is this piece of software a malware?
  - If so, what does the malware do?
    - Interesting behaviors (e.g., detection avoidance)
    - Information for repair/mitigation/takedown
    - Information about the business model

Static analysis

- Static reverse engineering
  - Disassemble, read the code, like in the lab
  - Would this work?
    - Obfuscation
    - Auto unpacking

Basics about binary executables

- Executable and Linkable Format
  - Text, data, rodata, bss
- Calling conventions
- Stack Layout
- Relocation
- Position-Independent Code (PIC)
- C++ internals

Dynamic analysis

- Execute the malware and observe its behaviors
- Challenges
  - How to contain/recover from damages?
  - How to trigger behaviors?

Sandboxes

- A (usually) virtualized execution environment to confine host damages
  - Emulators
  - OS-level sandboxes
  - Virtual machines

Arm race

- Countermeasures from malware authors
  - Is there a way to detect you're in a virtualized environment?
    - Instructions
    - OS environment
    - Network environment
  - If we know how malware detects, can we always fix?

State-of-the-art

- Bare metal analysis platform
    - How to recover?
- Countermeasures?
    - Environment-binding malware

Behavior monitoring

- Okay, suppose we have a good dynamic analysis environment, how do we know what kind of behaviors the analysis target does?
- Behaviors
  - Coarse-grained behaviors: OS-level behaviors
  - Fine-grained behaviors: function-level behaviors

OS-level monitoring

- OS refresh
    - Processes are isolated by OS
    - Modifications have to be done through system calls
- System call monitoring
    - Introspection

Traps and pitfalls

- Tal Garfinkel, *Traps and Pitfalls: Practical Problems in System Call Interposition Based Security Tools*
    - Incorrect replication/mirroring of OS state
    - Indirect paths
    - Race conditions
    - Incorrect subsetting of complex interfaces
    - Side-effects

Fine-grained tracing

- What kind of behaviors **cannot** be revealed at syscall level?
  - Countermeasures!!
    - Mutation engine (polymorphic/metamorphic)
    - Anti-analysis techniques
    - Domain name generation
    - etc

Fine-grained tracing (cont.)

- How?
  - Debugging
  - Emulators -> natively support
  - Hardware support

Triggers

- Malicious behaviors may only be revealed if certain preconditions are satisfied
- How to solve?
  - Decoys: typical targets of malware
  - Forced execution: not always doable

Network behaviors

- What if the malware tries to infect other machines?
  - Local network
  - Internet
- What if the malware tries to connect to C&C server?
  - How can you tell?
  - Allow or forbid?

Honeynet

- Two major components
  - Network decoys -> allow local infection
  - Gateway -> disallow Internet infection
    - Unless in whitelist

Malicious behaviors

- What kind of behaviors would cause the target to be classified as malware?
  - Replication, both locally and through network
  - Compromising the integrity of the OS
    - Autorun, rootkit, backdoor, etc
  - Leak the privacy of the users
  - Connecting to known malicious host or host of bad reputation
  - Monetization channels
    - Send spam, DDoS, premium SMS, AD fraud, fake AV, encryption, etc.

Make it scale

- Due to polymorphic and metamorphic, AV companies may collect millions of unique instances per day, how to make sure they are all analyzed?
    - **Automation!!**
- Limitations
    - Limited execution time
    - Only detects known malicious behaviors

By the way, how they collect samples?

- Exchange
- Client submissions
- Crawling
- Honeypot (worm-like malware)
- Honeyclient (drive-by downloads)

Infection cleanup

- Once malware detected on a system, how do we get rid of it?
- Restoring/repairing files (registry is also files)
  - Part of what AV companies sell
- Is there any guarantee?
  - What if there is a rootkit?
  - What if there is a bootkit?
  - What if the BIOS/firmware is infected?

"*nuke the entire site from orbit. It's the only way to be sure*"

- ALIENS

Network side detection: worm

- Can we detect worm traffics and block them?
- Idea #1: generate signature based on payload (exploits)
    - Issue? Polymorphic/metamorphic payload
- Idea #2: generate signature based on network behaviors
    - Works well for aggressive worms (code red, slammer)
    - Not so effective if malware tries to hide

Network side detection: botnet

- Can we take down the command & control servers?
- Q: how to bot connects to the C&C servers?
  - Hard coded IP addresses
  - Domain names (e.g., bot.net)
  - P2P

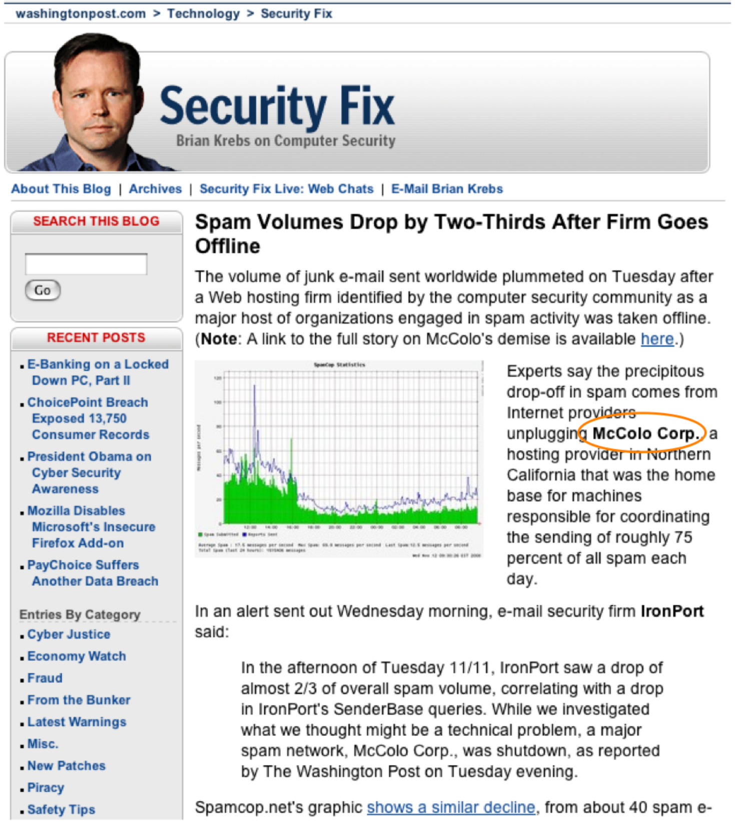
Network side detection: worm

- Can we detect worm traffics and block them?
- Idea #1: generate signature based on payload (exploits)
  - Issue? Polymorphic/metamorphic payload
- Idea #2: generate signature based on network behaviors
  - Works well for aggressive worms (code red, slammer)
  - Not so effective if malware tries to hide

Network side detection: botnet

- Can we take down the command & control servers?
- Q: how to bot connects to the C&C servers?
    - Hard coded IP addresses
    - Domain names (e.g., bot.net)
    - P2P

Taking down botnets: hosts



washingtonpost.com > Technology > Security Fix

Security Fix

Brian Krebs on Computer Security

About This Blog | Archives | Security Fix Live: Web Chats | E-Mail Brian Krebs

RECENT POSTS

- E-Banking on a Locked Down PC, Part II
- ChoicePoint Breach Exposed 13,750 Consumer Records
- President Obama on Cyber Security Awareness
- Mozilla Disables Microsoft's Insecure Firefox Add-on
- PayChoice Suffers Another Data Breach

Entries By Category

- Cyber Justice
- Economy Watch
- Fraud
- From the Bunker
- Latest Warnings
- Misc.
- New Patches
- Piracy
- Safety Tips

Spam Volumes Drop by Two-Thirds After Firm Goes Offline

The volume of junk e-mail sent worldwide plummeted on Tuesday after a Web hosting firm identified by the computer security community as a major host of organizations engaged in spam activity was taken offline. (**Note**: A link to the full story on McColo's demise is available here.)

Experts say the precipitous drop-off in spam comes from Internet providers unplugging **McColo Corp.**, a hosting provider in Northern California that was the home base for machines responsible for coordinating the sending of roughly 75 percent of all spam each day.

In an alert sent out Wednesday morning, e-mail security firm **IronPort** said:

> In the afternoon of Tuesday 11/11, IronPort saw a drop of almost 2/3 of overall spam volume, correlating with a drop in IronPort's SenderBase queries. While we investigated what we thought might be a technical problem, a major spam network, McColo Corp., was shutdown, as reported by The Washington Post on Tuesday evening.

Spamcop.net's graphic shows a similar decline, from about 40 spam e-

Arm race: host take down

- Botmaster countermeasures?
- Idea #1: keep moving around the master server
  - Use domain name instead of fixed IP addresses
  - Rapidly alter address associated w/ name (*fast flux*)
- Idea #2: buy off the host/ISP
  - **Bullet-proof hosting**

Taking down botnets: domain name

- Block/seize/sinkhole the domain name used by C&C servers
  - This is what's currently often used, often to good effect
  - May require court orders
- Botmaster countermeasures?
  - Register a large list of domain names and switch to a new one after a while (e.g., everyday)
  - How? **Domain Generation Algorithm** (DGA)
  - State-of-the-art

Arm race: peer-to-peer

- P2P networks: resolve name inside the own network
  - Distributed hash table (DHT)
  - Kademlia (BT, eMule)
- P2P botnets: locate C&C server through P2P network
  - Not really more resilient: rely on seeds to bootstrap
- Countermeasures
  - Machine learning based protocol detection

Arm race: steganography

- Use legitimate channel to send/receive commands
    - Twitter, Facebook, Google, etc
- Can also be used to fetch domain name, bootstrap seed

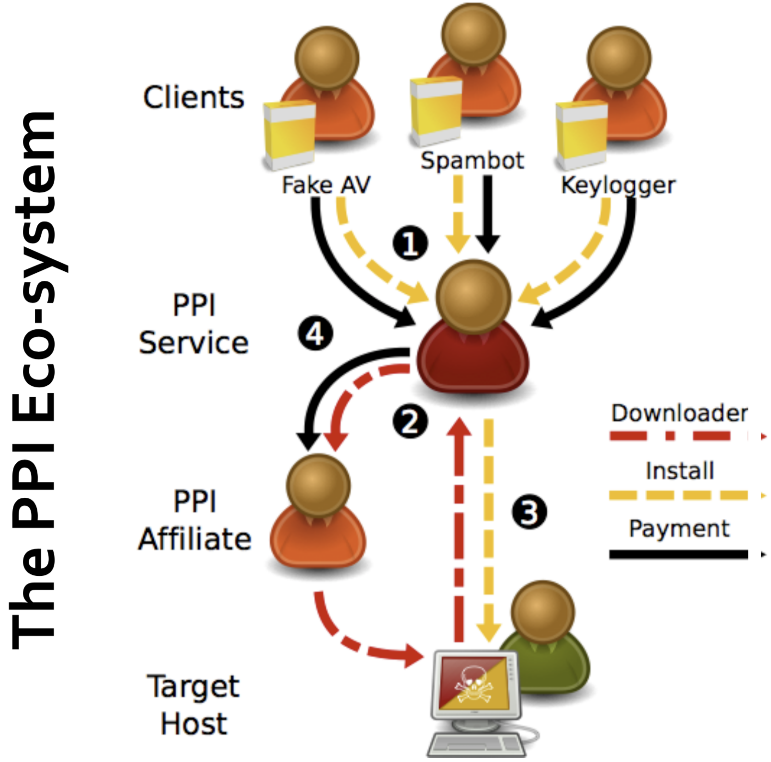
Two types of malware

- Two types of malware
  - Targeted (a.k.a. **advanced persistent threat**, APT), state-driven, high tech, highly stealthy
  - Large-scale infection, **monetization-driven**, low tech
- For the second type of malware, the most effective way to stop them is the economical way
  - **Cut their monetization channel**
- But we need to understand how they monetize first!

Understanding the underground economy

- What is their business model?
    - Where does the money come from?
    - How money flows?
- What is the criminal infrastructure?
    - Hosts, DNS provider, payment processor
- Goal: find the weakest link

Example: pay per install (PPI) ecosystem



The walled-garden model

- Why there are only a few malware on iOS devices?
    - How can you monetize on iOS?
    - How can you achieve large infection/installation?
- A healthy ecosystem matters a lot!