ENEE 457: Computer Systems Security
11/16/16

Lecture 21
Malware

Charalampos (Babis) Papamanthou

Department of Electrical and Computer Engineering
University of Maryland, College Park

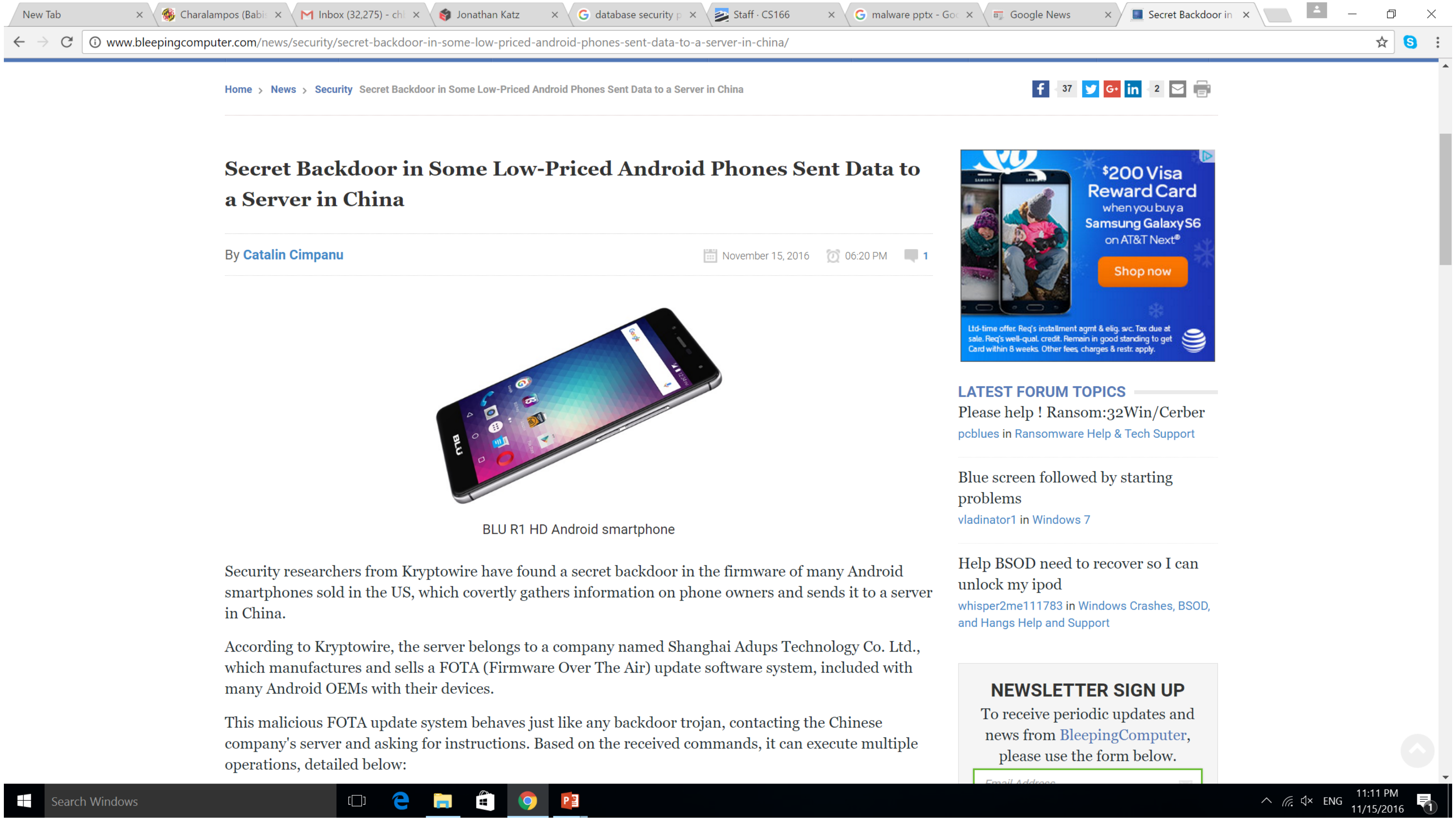
Type 1: Insider Attacks

- An **insider attack** is a security breach that is caused or facilitated by someone who is a part of the very organization that controls or builds the asset that should be protected.
- In the case of malware, an insider attack refers to a security hole that is created in a software system by one of its programmers.

Insider Attack 1: Backdoors

- A **backdoor,** which is also sometimes called a **trapdoor,** is a hidden feature or command in a program that allows a user to perform actions he or she would not normally be allowed to do.
- When used in a normal way, this program performs completely as expected and advertised.
- But if the hidden feature is activated, the program does something unexpected, often in violation of security policies, such as performing a privilege escalation.
- Benign example: **Login for a special user without password checking (maybe introduced during debugging)**

In the news



Home › News › Security Secret Backdoor in Some Low-Priced Android Phones Sent Data to a Server in China

Secret Backdoor in Some Low-Priced Android Phones Sent Data to a Server in China

By Catalin Cimpanu November 15, 2016 06:20 PM 1

BLU R1 HD Android smartphone

Security researchers from Kryptowire have found a secret backdoor in the firmware of many Android smartphones sold in the US, which covertly gathers information on phone owners and sends it to a server in China.

According to Kryptowire, the server belongs to a company named Shanghai Adups Technology Co. Ltd., which manufactures and sells a FOTA (Firmware Over The Air) update software system, included with many Android OEMs with their devices.

This malicious FOTA update system behaves just like any backdoor trojan, contacting the Chinese company's server and asking for instructions. Based on the received commands, it can execute multiple operations, detailed below:

LATEST FORUM TOPICS

Please help ! Ransom:32Win/Cerber
pcblues in Ransomware Help & Tech Support

Blue screen followed by starting problems
vladinator1 in Windows 7

Help BSOD need to recover so I can unlock my ipod
whisper2me111783 in Windows Crashes, BSOD, and Hangs Help and Support

NEWSLETTER SIGN UP

To receive periodic updates and news from BleepingComputer, please use the form below.

Email Address

Insider Attack 2: Logic Bombs

- A **logic bomb** is a program that performs a malicious action as a result of a certain logic condition.
- A classic example: A programmer codes up a payroll system so that it crashes if it processes two consecutive payrolls without paying him

Defenses against Insider Attacks

- Avoid single points of failure.
- Use code walk-throughs.
- Use archiving and reporting tools.
- Limit authority and permissions.
- Physically secure critical systems.
- Monitor employee behavior.
- Control software installations.

Type 2: Computer Viruses

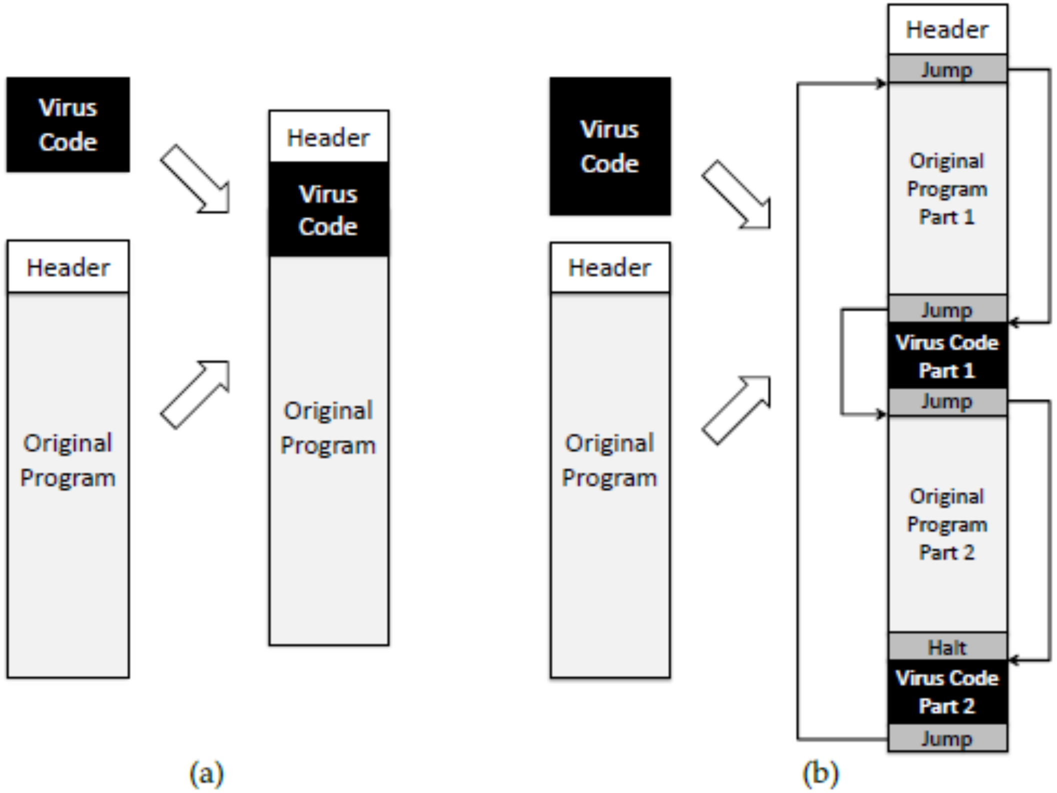
- A **computer virus** is a program that is able to copy itself when it is run. Very often, computer viruses are run as a part of other programs.
- This self-replication property is what distinguishes computer viruses from other kinds of malware, such as logic bombs.
- Another distinguishing property of a virus is that replication requires some type of **user assistance,** such as clicking on an email attachment or sharing a USB drive.

Virus Phases

- **Dormant phase**
- The virus program is idle during this stage. The virus program has managed to access the target user's computer or software, but during this stage, the virus does not take any action. The virus will eventually be activated by the "trigger" which states which event will execute the virus, such as a date, the presence of another program or file, the capacity of the disk exceeding some limit or the user taking a certain action (e.g., double-clicking on a certain icon, opening an e-mail, etc.). Not all viruses have this stage.
- **Propagation phase**
- The virus runs a search routine and starts propagating, that is multiplying and replicating itself. The virus places a copy of itself into other programs or into certain system areas on the disk. The copy may not be identical to the propagating version; viruses often "morph" or change to evade detection by IT professionals and anti-virus software. Each infected program will now contain a clone of the virus, which will itself enter a propagation phase.[43]
- **Triggering phase**
- A dormant virus moves into this phase when it is activated, and will now perform the function for which it was intended. The triggering phase can be caused by a variety of system events, including a count of the number of times that this copy of the virus has made copies of itself.
- **Execution phase**
- This is the actual work of the virus, where the "payload" will be released. It can be destructive such as deleting files on disk, crashing the system, or corrupting files or relatively harmless such as popping up humorous or political messages on screen.

Degrees of Complication

- Viruses have various degrees of complication in how they can insert themselves in computer code.

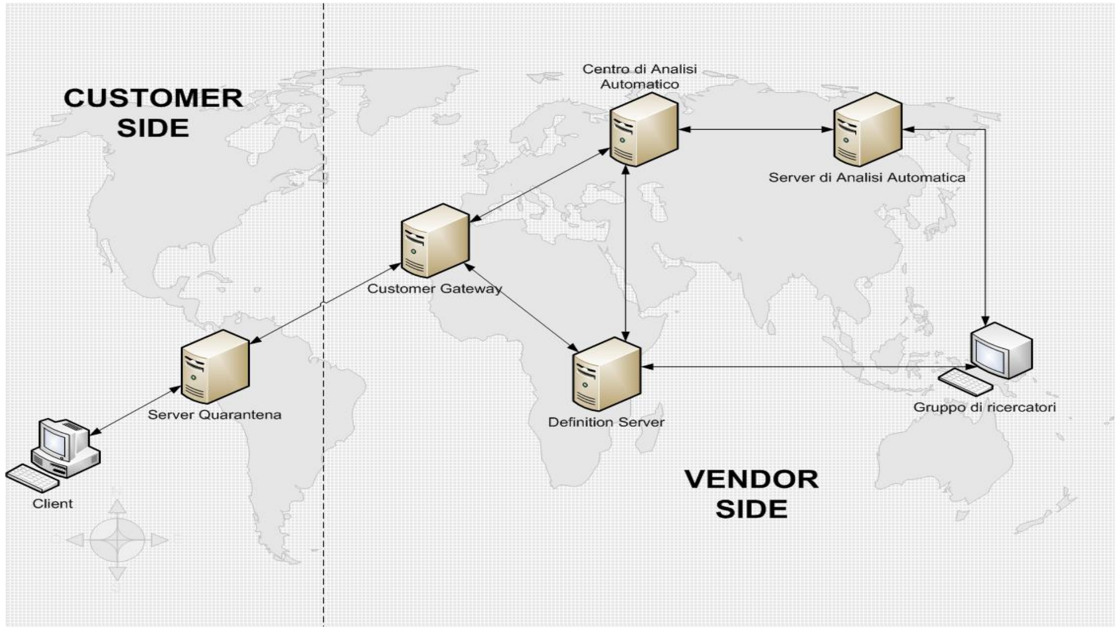


Signatures: A Malware Countermeasure

- A signature is a virus fingerprint
  - E.g., a string with a sequence of instructions specific for each virus
  - Different from a digital signature
- A file is infected if there is a signature inside its code
  - Fast pattern matching techniques to search for signatures
- All the signatures together create the malware database

Signatures Database

- Common Malware Enumeration (CME)
  - aims to provide unique, common identifiers to new virus threats
  - Hosted by MITRE
  - http://cme.mitre.org/data/list.html
- Digital Immune System (DIS)
  - Create automatically new signatures

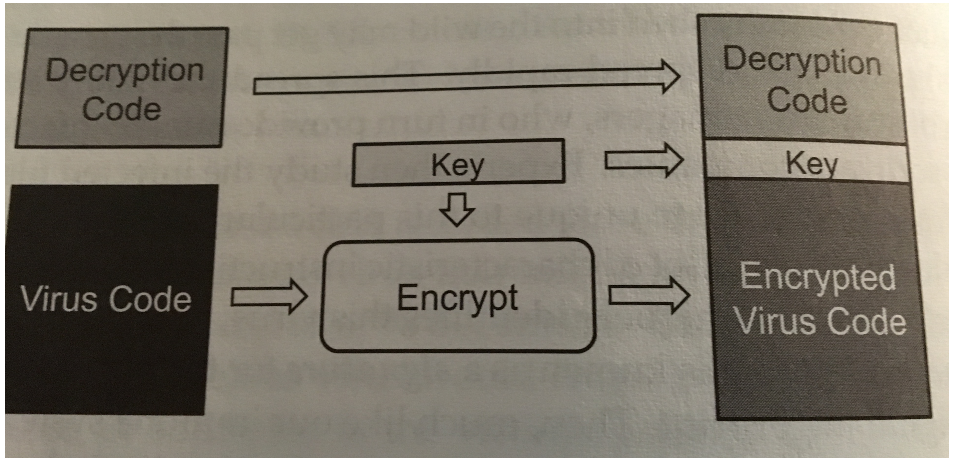


How to hide virus code?

- Encrypted Viruses
- Polymorphic Viruses
- Metamorphic Viruses

Encrypted Virus

- Encrypted virus
  - Decryption engine + encrypted body
  - Randomly generate encryption key
  - Detection looks for decryption engine



Encrypted Virus

- Polymorphic virus
    - Each body of the virus is encrypted using a different key
    - First copy of the virus uses a certain key/encryption method
    - Second copy of the virus uses different key/encryption method
- Metamorphic virus
    - Obfuscate code
    - Approaches include instruction reordering and inclusion of useless instructions
    - Challenging to detect

Obfuscated program

```
main(_){_^448&&main(-~_);
putchar(--_%64?32|-~7[__TIME__-
_/8%8][">'txiZ^(~z?"-48]>>";
;;====~$::199"[_*2&8|_/64]/(_&2
?1:8)%8&1:10);}
```

- Compile it and run it
- It is going to print the current time!

Type 3: Computer Worms

- A **computer worm** is a malware program that spreads copies of itself without the need to inject itself in other programs, and usually without human interaction.
- Thus, computer worms are technically not computer viruses (since they don't infect other programs), but some people nevertheless confuse the terms, since both spread by self-replication.
- In most cases, a computer worm will carry a malicious payload, such as deleting files or installing a backdoor.

Early History

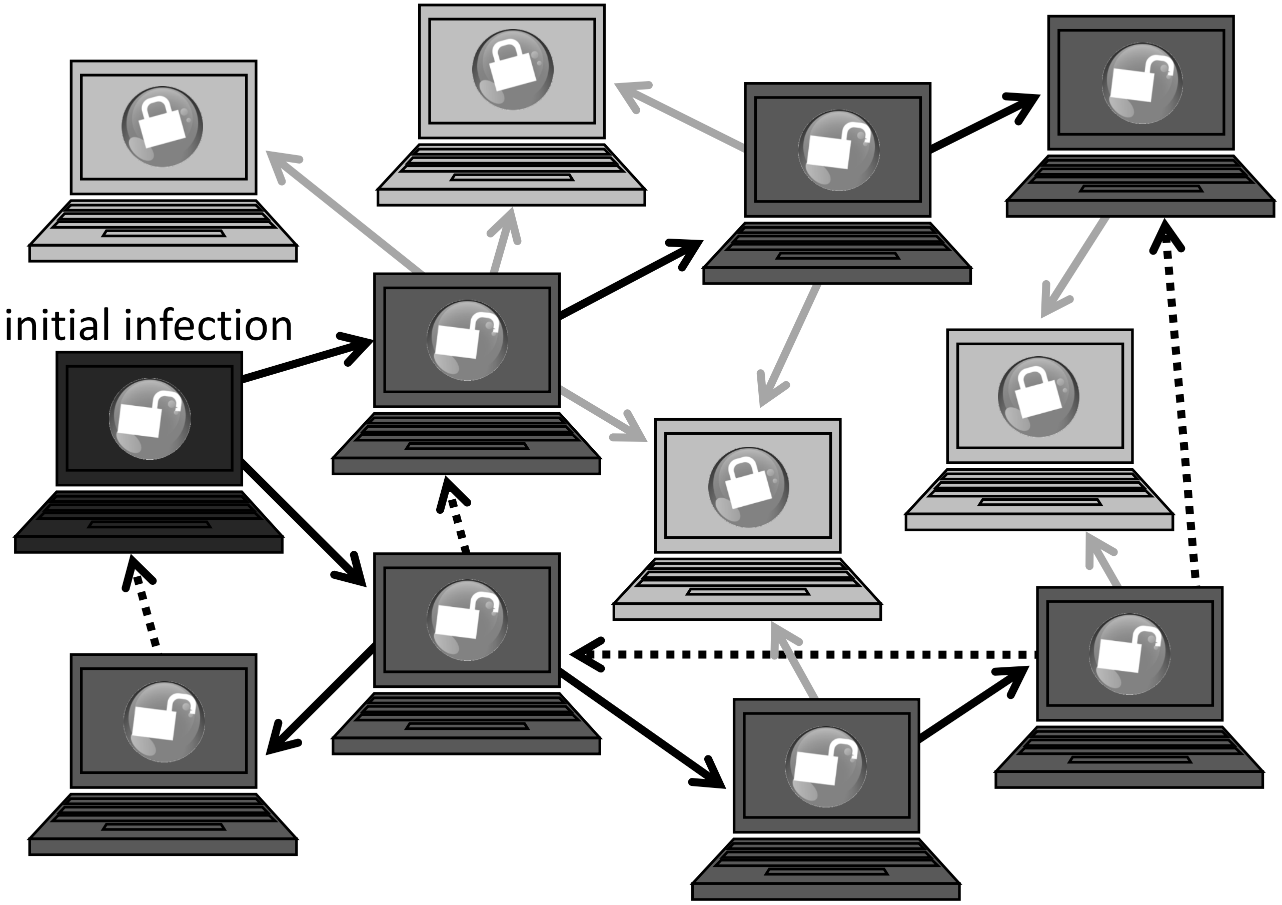
- First worms built in the labs of John Shock and Jon Hepps at Xerox PARC in the early 80s
- CHRISTMA EXEC, released in December 1987 was targeting IBM VM/CMS systems and was the first worm to use e-mail service
- The first internet worm was the Morris Worm, written by Cornell student Robert Tappan Morris and released on November 2, 1988

Worm Development

- Identify vulnerability still unpatched
  - exploit(`host`)
    - Go to `host`
    - Run the attack (e.g., delete files)
    - For `host1`, `host2`,…,`hostk` susceptible and not infected
      - Do exploit(`hosti`)
- Distributed graph search algorithm

Worm Propagation

- Worms propagate by finding and infecting vulnerable hosts.
  - They need a way to tell if a host is vulnerable
  - They need a way to tell if a host is already infected.



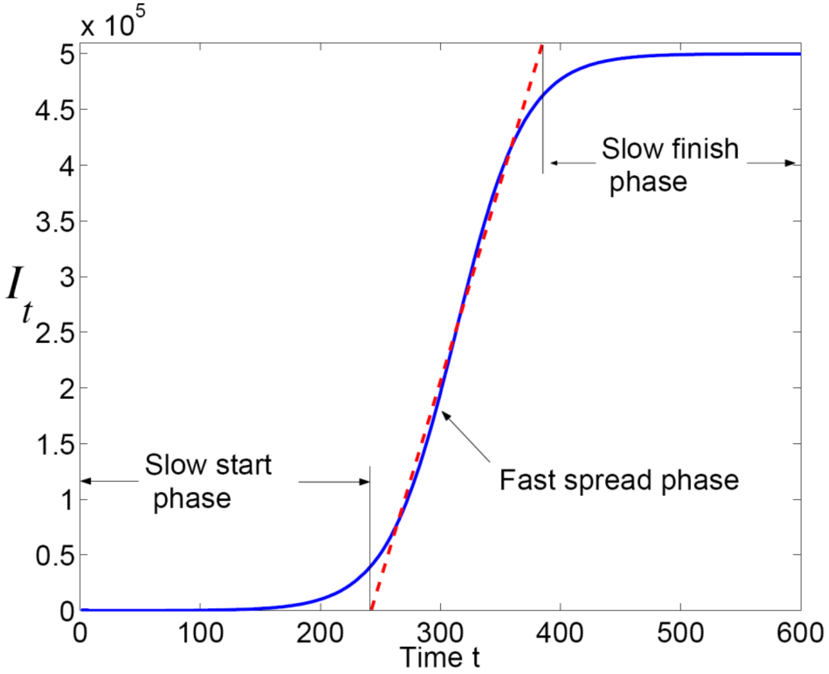
Propagation: Theory

- Classic epidemic model
  - ***N***: total number of vulnerable hosts
  - ***I***(***t***): number of infected hosts at time ***t***
  - ***S***(***t***): number of susceptible hosts at time *t*
  - ***I***(***t***) + ***S***(***t***) = ***N***
  - β: infection rate
- Differential equation for ***I***(***t***):

$$\mathbf{d}\boldsymbol{I(t)}/\mathbf{d}\boldsymbol{t} = \beta \boldsymbol{I}(\boldsymbol{t})\ \boldsymbol{S}(\boldsymbol{t})$$

- More accurate models adjust propagation rate over time

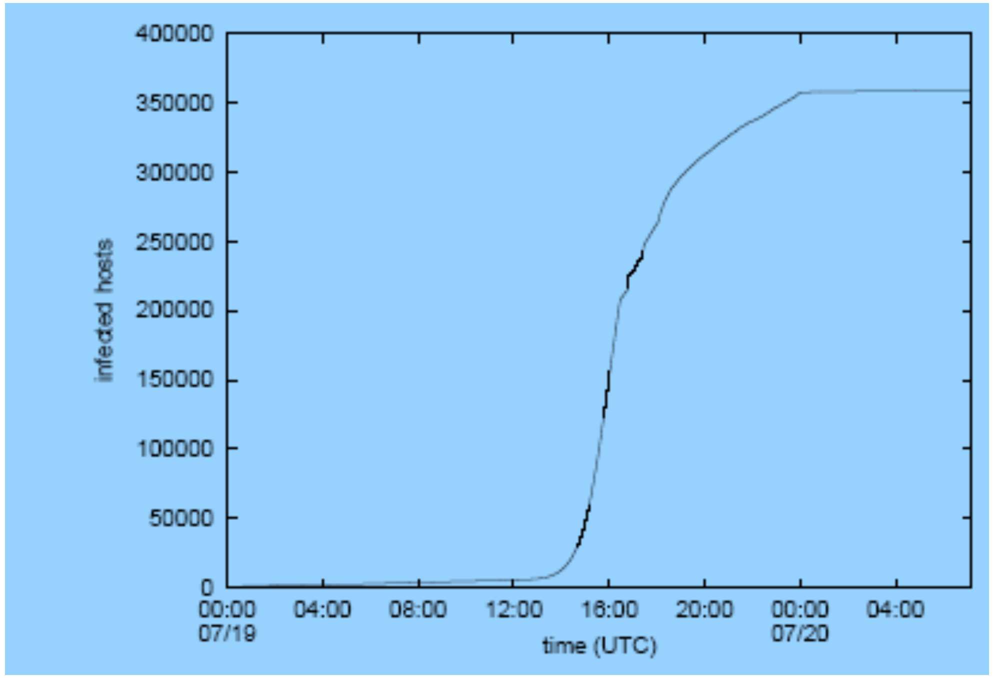
Source:
Cliff C. Zou, Weibo Gong, Don Towsley, and Lixin Gao. The Monitoring and Early Detection of Internet Worms, IEEE/ACM Transactions on Networking, 2005.



Propagation: Practice

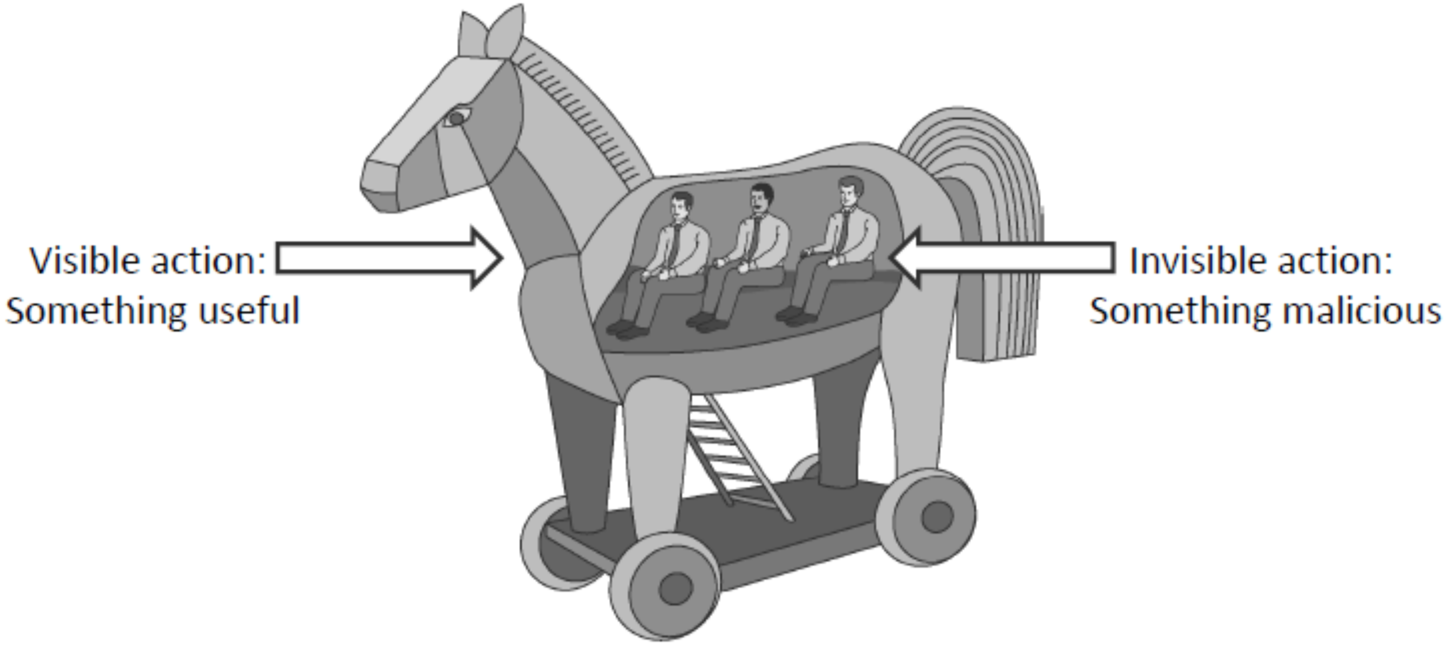
- Cumulative total of unique IP addresses infected by the first outbreak of Code-RedI v2 on July 19-20, 2001

Source:
David Moore, Colleen Shannon, and Jeffery Brown. Code-Red: a case study on the spread and victims of an Internet worm, CAIDA, 2002



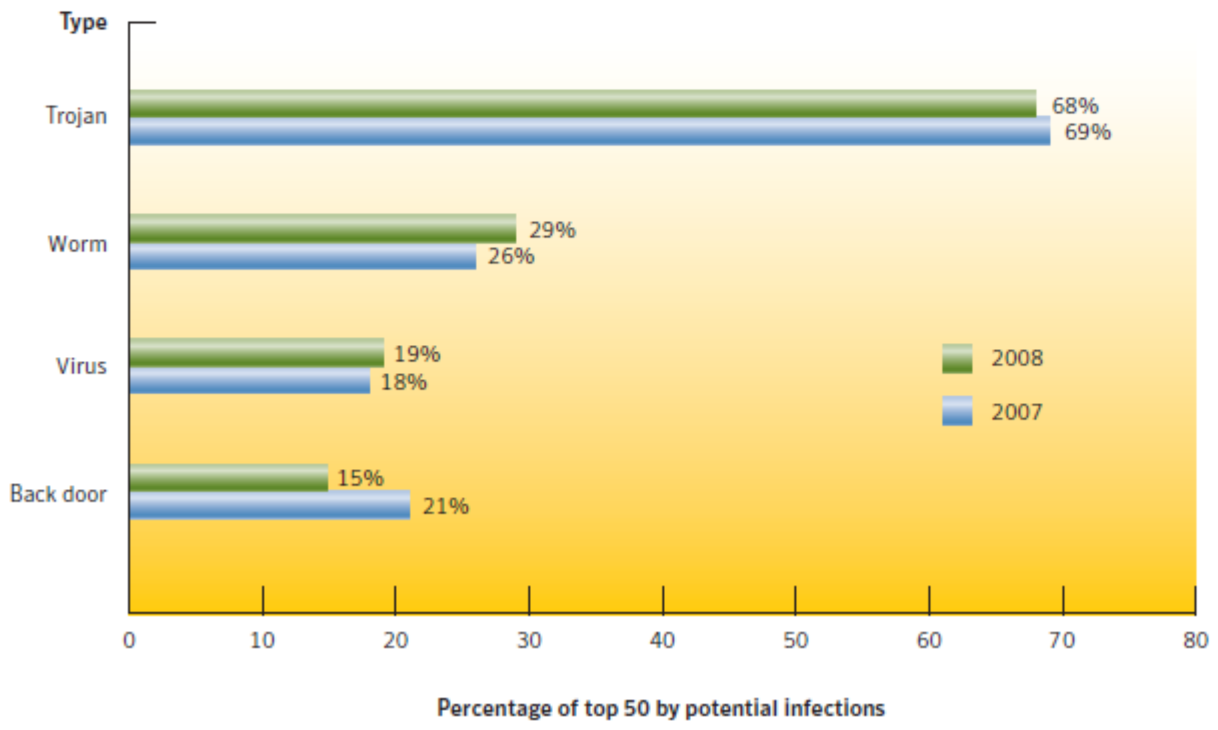
Trojan Horses

- A **Trojan horse (or Trojan)** is a malware program that appears to perform some useful task, but which also does something with negative consequences (e.g., launches a keylogger).
- Trojan horses can be installed as part of the payload of other malware but are often installed by a user or administrator, either deliberately or accidentally.



Current Trends

- Trojans currently have largest infection potential
  - Often exploit browser vulnerabilities
  - Typically used to download other malware in multi-stage attacks



Source:
Symantec Internet Security Threat Report, April 2009

Rootkits

- A rootkit modifies the operating system to hide its existence
  - E.g., modifies file system exploration utilities
  - Hard to detect using software that relies on the OS itself
- RootkitRevealer
  - By Bryce Cogswell and Mark Russinovich (Sysinternals)
  - Two scans of file system
  - High-level scan using the Windows API
  - Raw scan using disk access methods
  - Discrepancy reveals presence of rootkit
  - Could be defeated by rootkit that intercepts and modifies results of raw scan operations

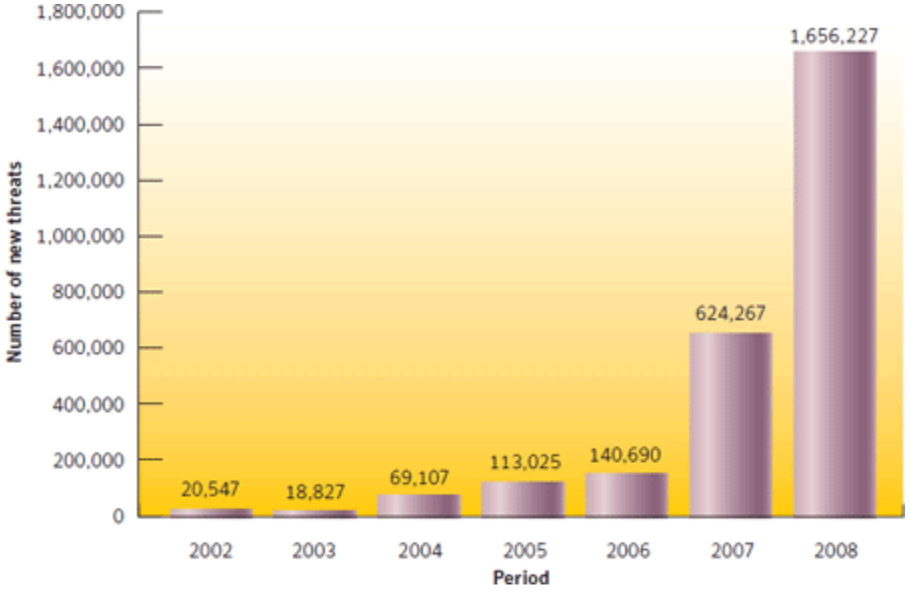
Zombie

- A zombie is a program that secretly takes over another Internet-attached computer and then uses it to launch attacks
- Used for launching denial of service attacks
- Can replicate

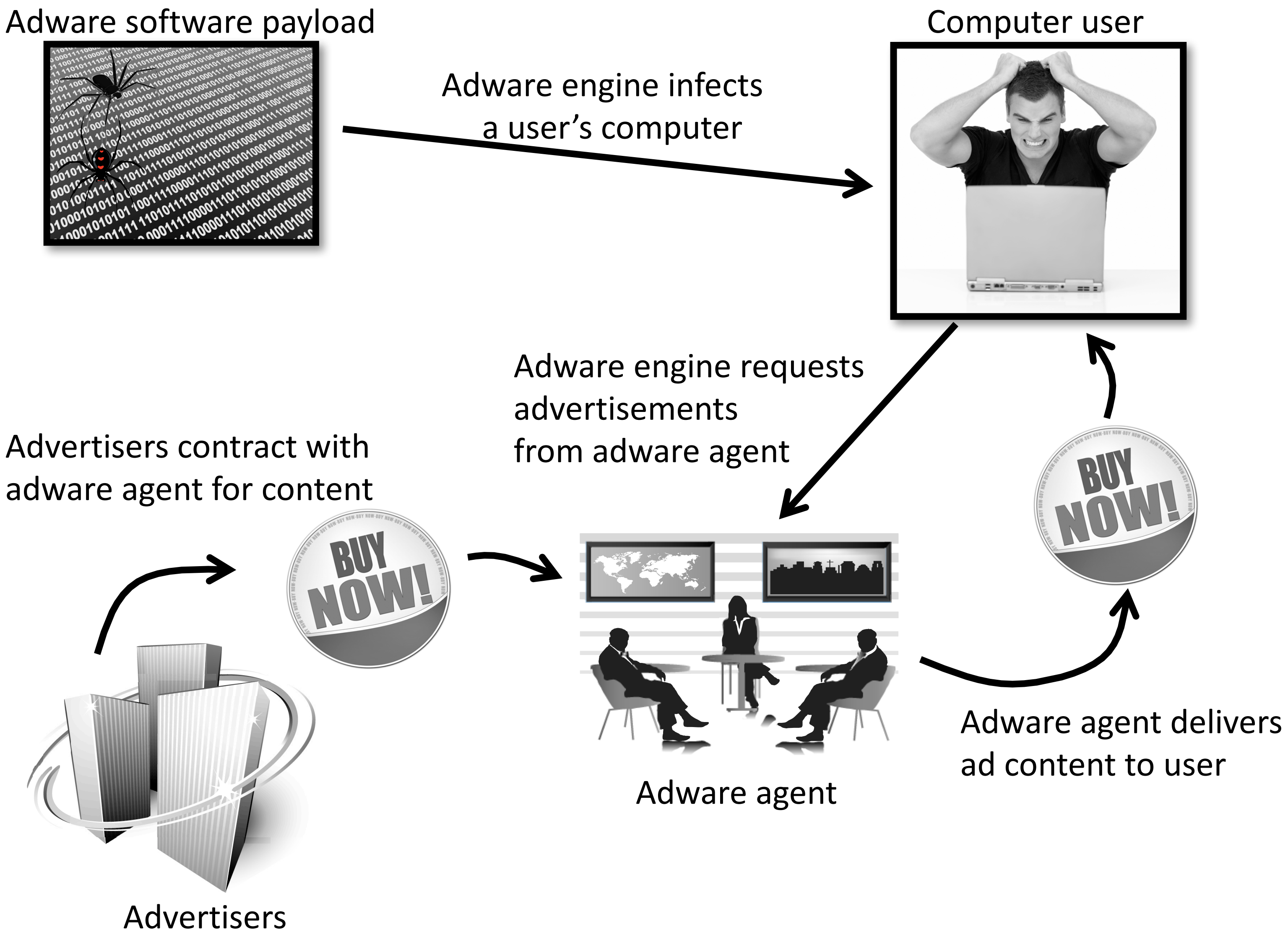
Economics of Malware

- New malware threats have grown from 20K to 1.7M in the period 2002-2008
- Most of the growth has been from 2006 to 2008
- Number of new threats per year appears to be growing an exponential rate.

Source: Symantec Internet Security Threat Report, April 2009



Adware



Adware software payload

Computer user

Adware engine infects a user’s computer

Adware engine requests advertisements from adware agent

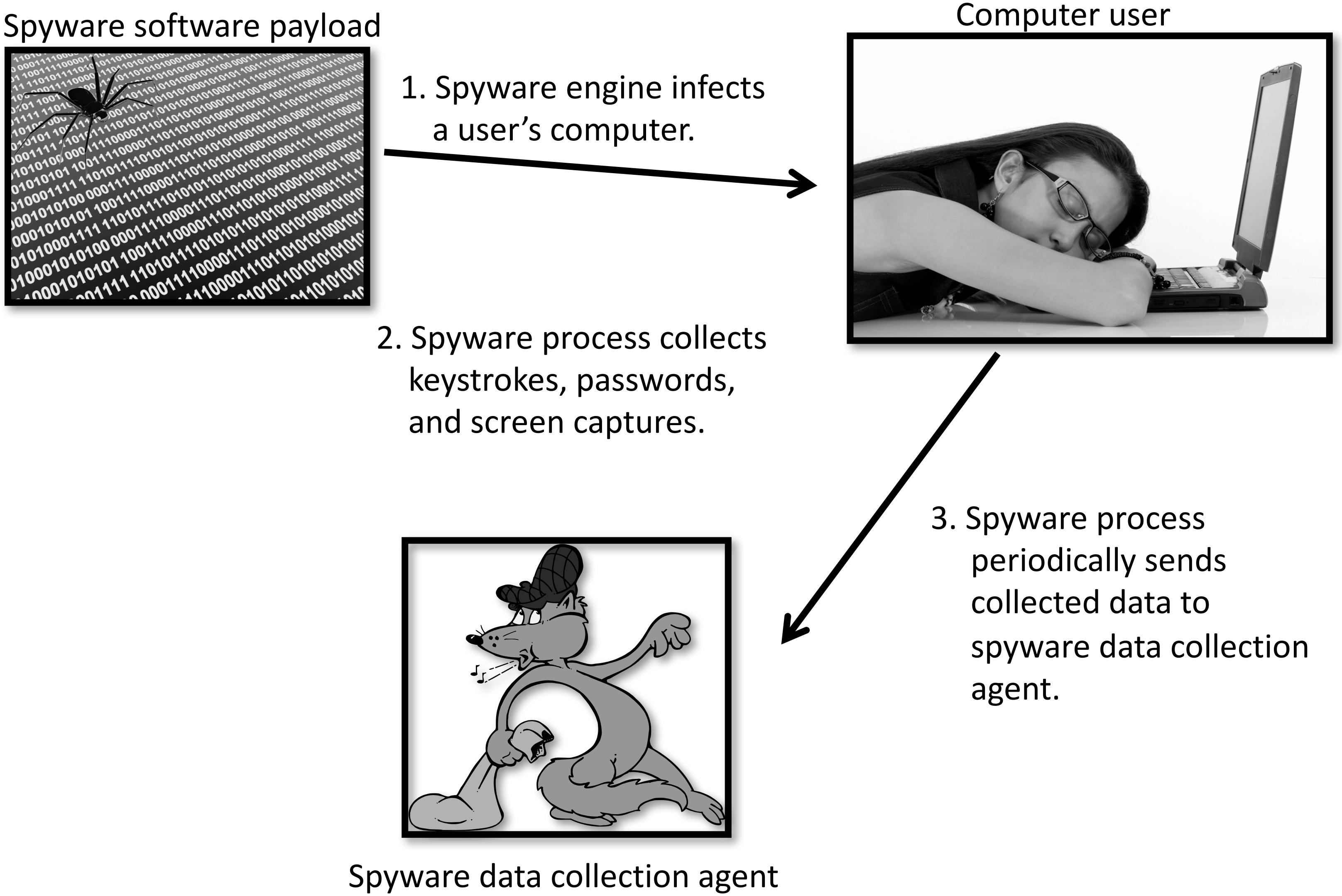
Advertisers contract with adware agent for content

Adware agent

Adware agent delivers ad content to user

Advertisers

Spyware



Spyware software payload

1. Spyware engine infects a user's computer.

Computer user

2. Spyware process collects keystrokes, passwords, and screen captures.

3. Spyware process periodically sends collected data to spyware data collection agent.

Spyware data collection agent

Countermeasures: White/Black Listing

- Maintain database of cryptographic hashes for
  - Operating system files
  - Popular applications
  - Known infected files
- Compute hash of each file
- Look up into database
- Needs to protect the integrity of the database

Online vs Offline Anti Virus Software

Online

- Free browser plug-in
- Authentication through third party certificate (i.e. VeriSign)
- Software and signatures update at each scan
- Poorly configurable
- Scan needs internet connection
- Report collected by the company that offers the service

Offline

- Paid annual subscription
- Installed on the OS
- Software distributed securely by the vendor online or a retailer
- Scheduled software and signatures updates
- Easily configurable
- Scan without internet connection
- Report collected locally and may be sent to vendor

Virus Detection is Undecidable

- Theoretical result by Fred Cohen (1987)
- Virus abstractly modeled as program that eventually executes infect
- Code for infect may be generated at runtime
- Proof by contradiction similar to that of the halting problem

- Suppose program isVirus(P) determines whether program P is a virus
- Define new program Q as follows:

```
if (not isVirus(Q))
   infect
stop
```

- Running isVirus on Q achieves a contradiction

Review

- Malware can be classified into several categories, depending on propagation and concealment
- Propagation
  - Virus: human-assisted propagation and infecting other programs
  - Worm: automatic propagation without human assistance
- Concealment
  - Rootkit: modifies operating system to hide its existence
  - Trojan: provides desirable functionality but hides malicious operation
- Various types of payloads, ranging from annoyance to crime