Special Topics in Security
ECE 5968

Engin Kirda
ek@ccs.neu.edu

Northeastern University

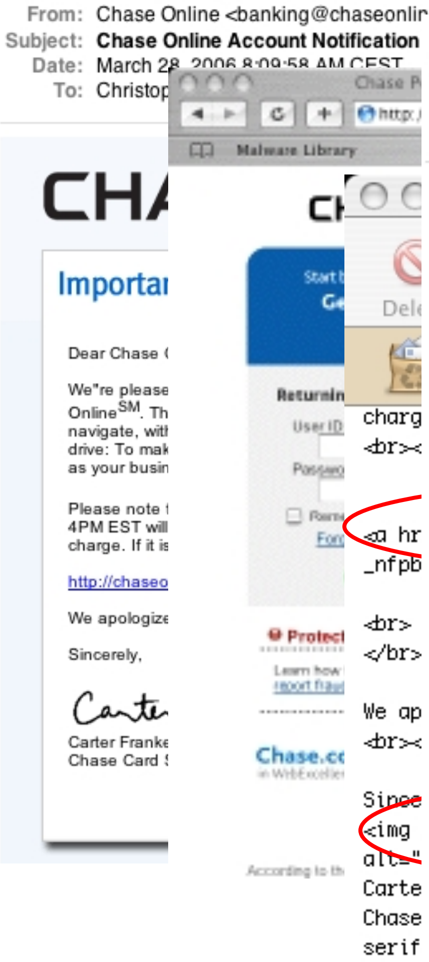
Internet Services Security (continued)

Phishing

- More recent scam that
  - exploits weakness of SMTP protocol and
  - social engineering aspects
- Tricks people into providing sensitive information
  - create a situation that asks receiver to act on (urgent) problem
  - provide a link to site to solve problem
  - site prepared by attacker
    - appearance of site is spoofed
    - asks for personal information

- Interesting side note
  - scammers typically require people to launder money
  - additional spam mails that invite people to "earn money with their bank account"

Phishing



From: Chase Online <banking@chaseonlin

Subject: **Chase Online Account Notification**

Date: March 28, 2006 8:09:58 AM CEST

To: Christop

CHA

Importa

Dear Chase

We"re please
Online^{SM}. Th
navigate, wit
drive: To mak
as your busin

Please note
4PM EST will
charge. If it is

http://chaseo

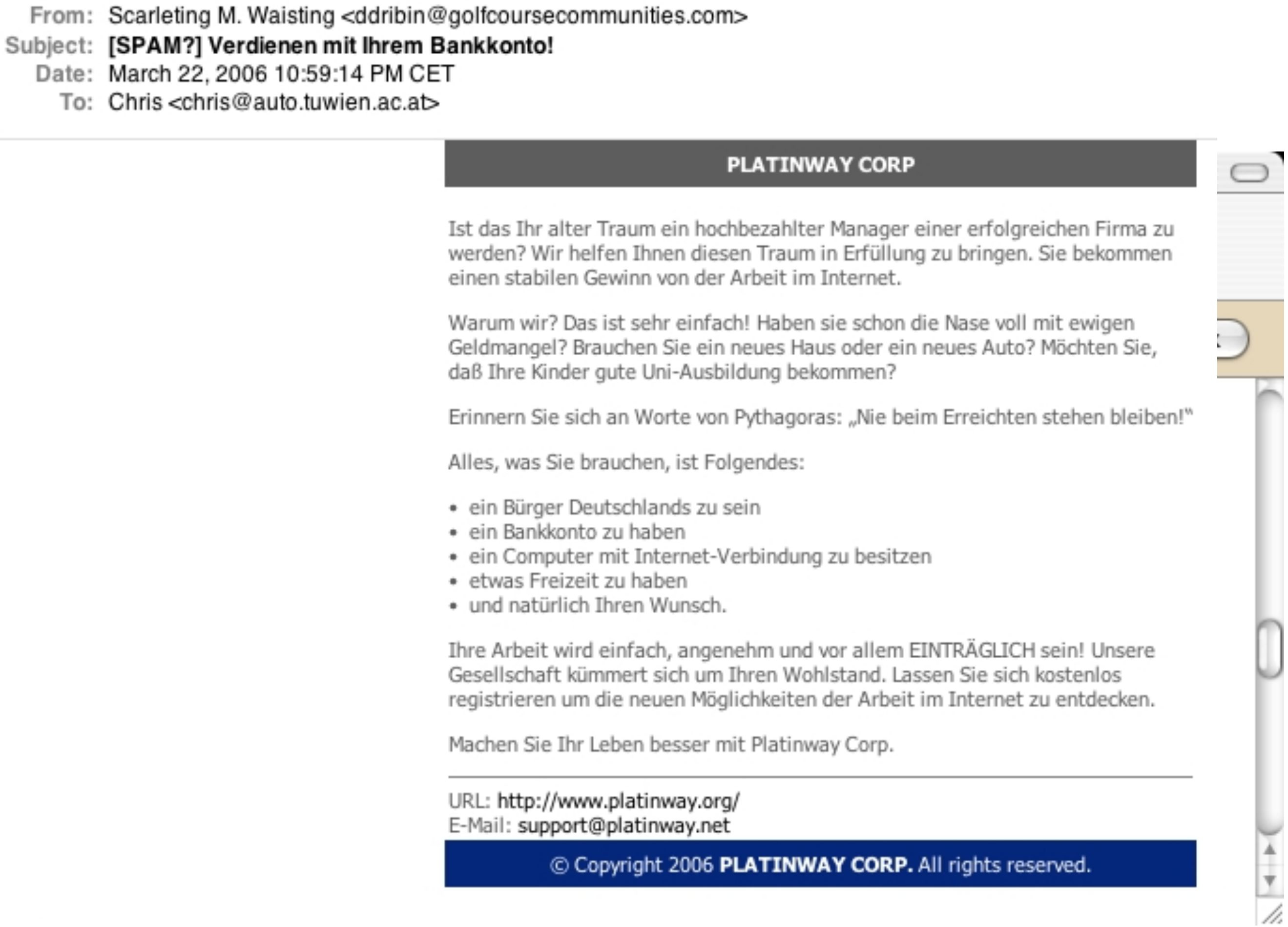
We apologize

Sincerely,

Carte

Carter Franke
Chase Card

```
charg
<br><
<a hr
_nfpb
<br>
</br>
We ap
<br><
Since
<img
alt="
Carte
Chase
serif
```



From: Scarleting M. Waisting <ddribin@golfcoursecommunities.com>

Subject: **[SPAM?] Verdienen mit Ihrem Bankkonto!**

Date: March 22, 2006 10:59:14 PM CET

To: Chris <chris@auto.tuwien.ac.at>

PLATINWAY CORP

Ist das Ihr alter Traum ein hochbezahlter Manager einer erfolgreichen Firma zu werden? Wir helfen Ihnen diesen Traum in Erfüllung zu bringen. Sie bekommen einen stabilen Gewinn von der Arbeit im Internet.

Warum wir? Das ist sehr einfach! Haben sie schon die Nase voll mit ewigen Geldmangel? Brauchen Sie ein neues Haus oder ein neues Auto? Möchten Sie, daß Ihre Kinder gute Uni-Ausbildung bekommen?

Erinnern Sie sich an Worte von Pythagoras: „Nie beim Erreichten stehen bleiben!"

Alles, was Sie brauchen, ist Folgendes:

- ein Bürger Deutschlands zu sein
- ein Bankkonto zu haben
- ein Computer mit Internet-Verbindung zu besitzen
- etwas Freizeit zu haben
- und natürlich Ihren Wunsch.

Ihre Arbeit wird einfach, angenehm und vor allem EINTRÄGLICH sein! Unsere Gesellschaft kümmert sich um Ihren Wohlstand. Lassen Sie sich kostenlos registrieren um die neuen Möglichkeiten der Arbeit im Internet zu entdecken.

Machen Sie Ihr Leben besser mit Platinway Corp.

URL: http://www.platinway.org/
E-Mail: support@platinway.net

© Copyright 2006 **PLATINWAY CORP.** All rights reserved.

Phishing

- Camouflage techniques
  - use images from original site
  - sender name and email addresses can be faked easily
  - attempt to avoid obvious spelling and grammar mistakes :-)

  - link to phishing site must be obfuscated
  - URL and port redirection

    `http://www.bank.com@evil.com:80/index.html`

  - UnDotted IP addresses
    - 32-bit value used as address without any dots
    - Could by-pass Internet Explorer security settings

Phishing Defense

- User education
  - Question: Will phishing remain a problem in 10 years from now?

- Stronger authentication of sources
  - difficult without global PKI
  - ad-hoc mechanisms such as SiteKey or iTans
    - can be bypassed by active phishing attacks

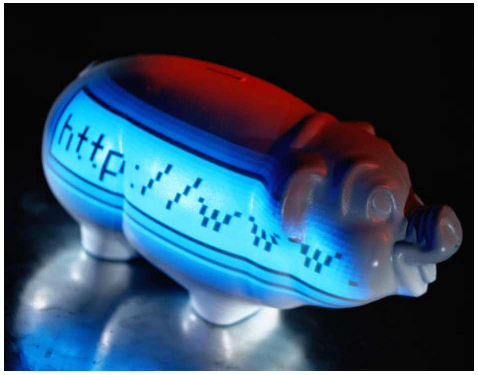
- Techniques to detect sites that faithfully mimic others
  - SpoofGuard
    - browser plug-in
    - uses heuristics such as image similarity, domain name similarity, …
  - active crawling of the web for suspicious sites

Phishing Defense

- Techniques to ensure that password is not shared between sites
  - problem that users want to reuse passwords
- Password hashing
  - generate unique passwords for different sites
    - combine original password and URL
  - cannot protect sensitive information in general, because data changed
- AntiPhish
  - browser plug-in for Firefox and Internet Explorer
  - user explicitly tags of all sensitive information
  - sharing of information results in warnings

- Distributed solutions
  - reuse of information is submitted to central server that can aggregate
  - spike of reuse for a particular domain is suspicious

Pharming



- Idea (and name) similar to phishing
- DNS entry of victim organization is hijacked
- Clients are redirected to server of attacker
  - e.g., New York ISP provider Panix in January ‘05
- Sometimes, DNS entries can be hijacked by simply calling up the registrar
  - Secure email provider Hushmail in April ‘05

Malware (Malicious Code)

The type of threat…

- … Is often not too easy to determine
  - Even given these loose definitions of malware types
  - A threat might be hybrid
    - e.g., a Trojan might also be spyware at the same time
    - A worm might propagate over the network by having Trojan functionality
- A *blended threat* is when a virus exploits a technical vulnerability to propagate itself
- I left out one malware category, can you name this category? ;)
  - Botnets! ☺ More on this in later lectures

Naming

- As malware spreads, the main concern is to catch it
  - A second concern is to give it a name
  - Naming is important for companies because of marketing reasons
  - There is no central naming authority
  - A piece of malware often has different names, depending on who is detecting it
  - Will there be standardization soon?
    - Probably not – too much malware
    - Malware might change fast so naming standards are difficult to establish

Naming

- Here is a case of the *same* malware instance as name by different vendors…

Bagle.C

Email-worm.Win32.Bagle.c

W32/Bagle.c@MM

W32.Beagle.C@mm

WORM.BAGLE.C

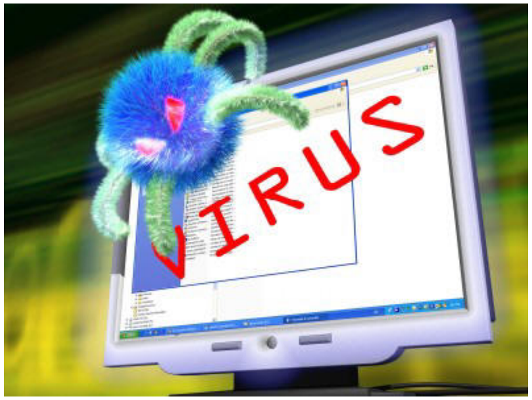
Worm.Bagle.A3

Malware Authorship

- People whose machines have been infected…
  - May have more colorful terms to describe person who created malware
  - e.g., a*$$!*!!! ;)
  - Common terms are *malware author*, *malware writer*, and *virus writer*
- There is a distinction made between writing and distributing
  - Based on our terminology before, is writing malware hacking?
  - Yes and no… malware attacks are largely automated, whereas hacking tends to be more manual

Viruses

- A virus has three components
  - Infection mechanism
    - How a virus spreads
    - The exact means through which virus spreads is called…
    - An *infection vector*
    - What if a virus infects in multiple ways?
    - *multipartite*
  - Trigger
    - Deciding whether to deliver or not
  - Payload
    - What the virus does



Viruses

- In pseudo code, a virus would like look this

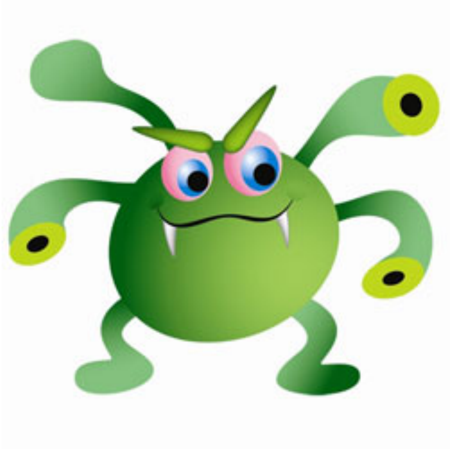
```
def virus:
    infect()
    if trigger() is true:
            payload()
```

Question: Remember, this is a virus. What does the infection function do?

Viruses

- Generally, *k* targets may be infected each time the infection routine is run

```
def infect:
    repeat k times:
      target = select_target()
      if no target:
            return
     infect_code(target)
```



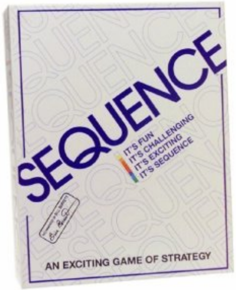
Question: What is the tricky part of the code here?

→ select_target() – virus does not want to infect code multiple times

Virus Classification by Target

- A popular way to classify viruses is by looking at what they try to infect
  - We will first look at three classes of viruses: boot-sector viruses, executable file infectors, and data file infectors (i.e., called macro viruses)

Question: Who can tell me what these types of viruses are?



Boot Sequence

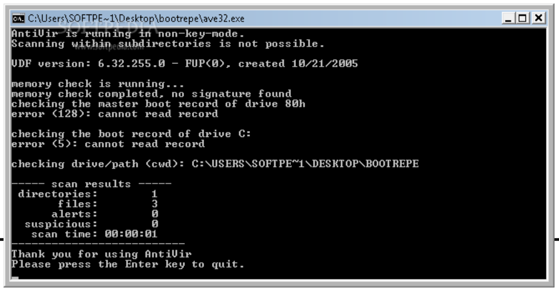
- Boot sequence on most machines typically goes through these steps

  1.) Power on

  2.) ROM-based instructions run, performing a self-test, device detection, and initialization. The boot device is identified and boot block read from it

  3.) After boot block is loaded, control is transferred to loaded code → *primary boot*

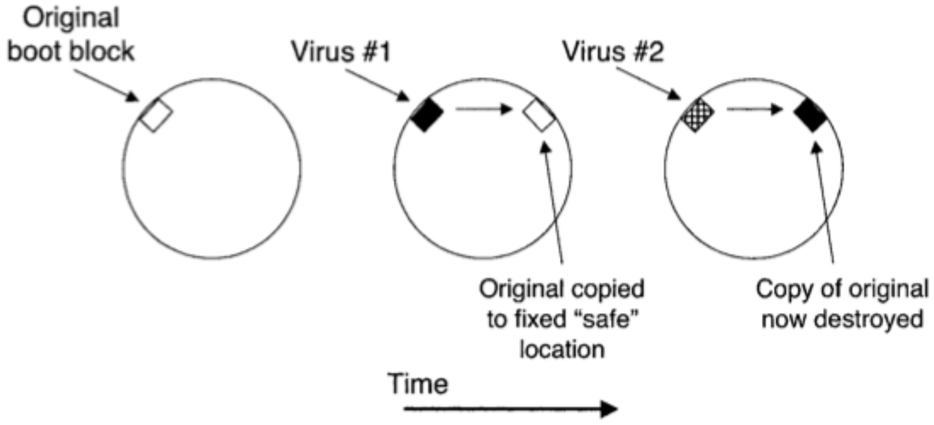
  4.) The loaded code loads a larger, more sophisticated code that understands file structure, and transfers control → *secondary boot*



```
AntiVir is running in non-key-mode.
Scanning within subdirectories is not possible.

VDF version: 6.32.255.0 - FUP(0), created 10/21/2005

memory check is running...
memory check completed, no signature found
checking the master boot record of drive 80h
error (128): cannot read record

checking the boot record of drive C:
error (5): cannot read record

checking drive/path (cwd): C:\USERS\SOFTPE~1\DESKTOP\BOOTREPE

----- scan results -----
 directories:        1
       files:        3
      alerts:        0
  suspicious:        0
   scan time: 00:00:01
--------------------------
Thank you for using AntiVir
Please press the Enter key to quit.
_
```

Boot-Sector Infector

- A virus that infects by copying itself to the boot block
  - Question: What happens to the original boot block?
  - The issue with moving the boot block is that disk space needs to be allocated, and much code is needed
  - Hence, many viruses tended to copy the block to the same location (e.g., Stoned and Michelangelo)
  - Question: What is the problem with that?


Original boot block
Virus #1
Virus #2
Original copied to fixed "safe" location
Copy of original now destroyed
Time

[Aycock06]

Boot-Sector Infector

- In general, infecting boot sector is strategically sound
  - The virus is loaded *before* any AV software
  - BSIs used to be rare, but now, new malware that has boot-sector functionality has been introduced (i.e., stoned bootkit)
  - Question: Suppose that you were creating a protection technique against boot sector infectors, what would your solution be? ;)
    - Many BIOS instances have a boot block protection that can be enabled
    - Authorization is required

File Infectors

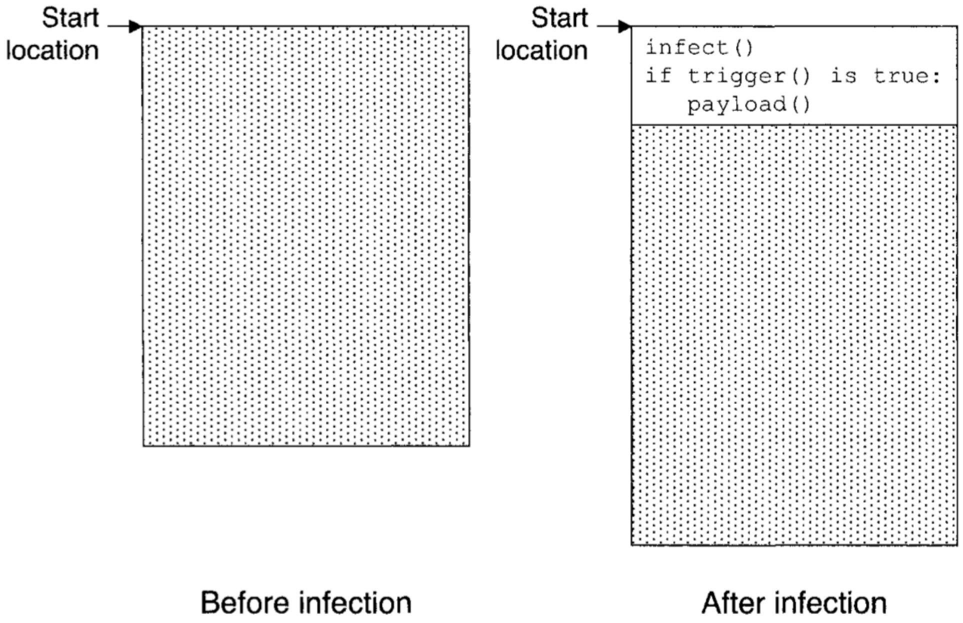
- Operating systems have a notion of files that are executable
  - In a broader sense, executable files may also include files that can be run by a command-line user "shell"
  - A file infector infects files that are executable (e.g., including batch files and shell scripts) – binary files are the most popular
  - Two main issues

1.) *Where is the virus placed?*

2.) *How is the virus executed when the infected file is run?*

Insertion: Beginning of the File

- Older, simple executable file formats (e.g., .COM in MSDOS) would treat the entire file as code and data
  - The entire file would be loaded into memory and execution would jump to the beginning



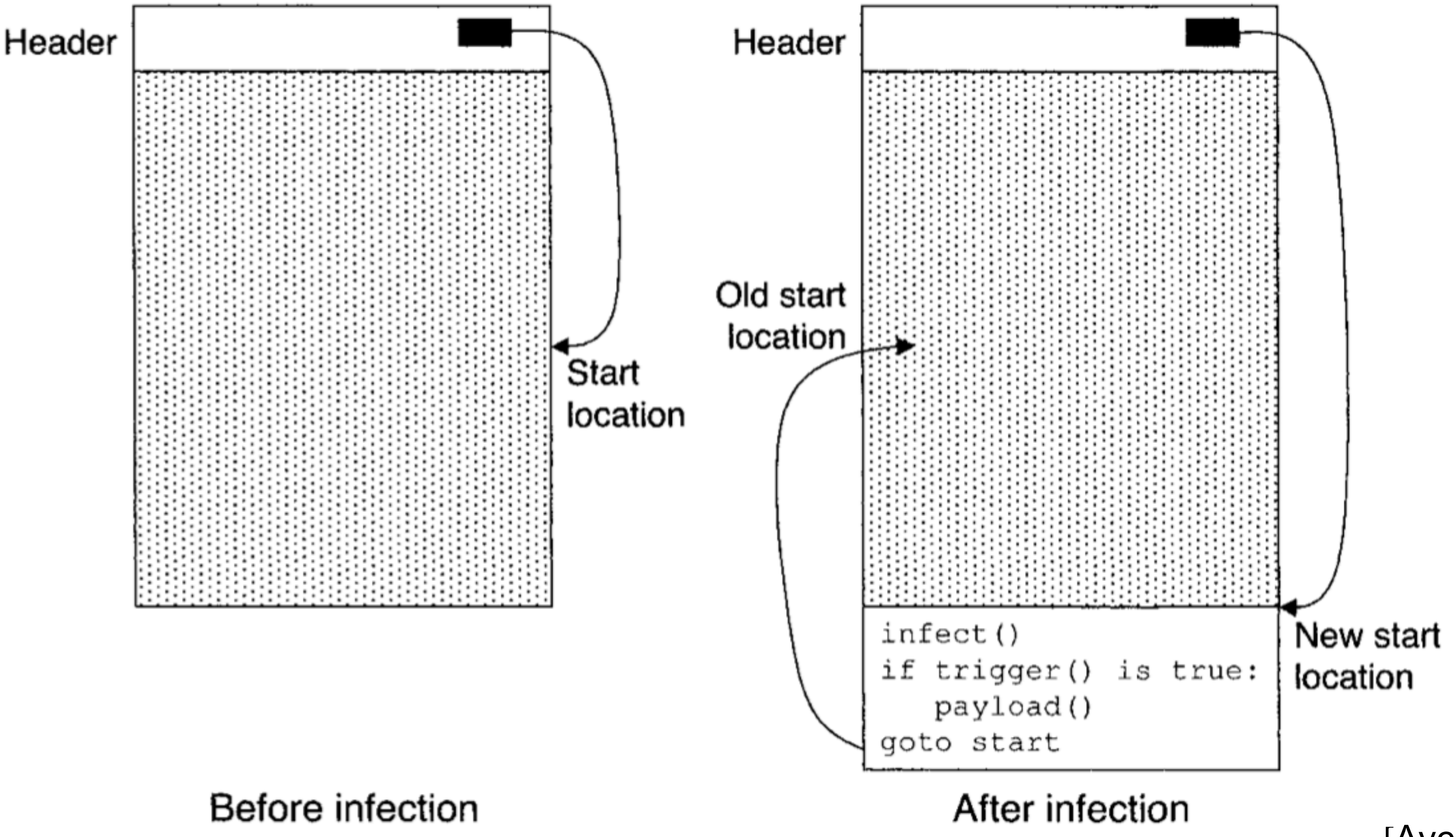
```
infect()
if trigger() is true:
   payload()
```

[Aycock06]

Insertion: End of the File

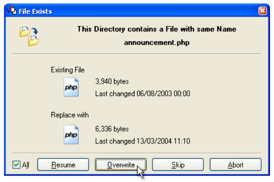
- Question: What is easier, appending to the end or to the beginning of a file?
- How does the virus get control?
  - The original instructions in the code can be saved (somewhere) and replaced by a jump to viral code. After execution, virus transfers control back to infected code. The code can be run in new location, or may be restored
  - Many executable file formats specify a start location. Virus may change this, store the original value, and jump to it after it’s done

Insertion: End of the File



```
infect()
if trigger() is true:
   payload()
goto start
```

[Aycock06]



File Exists

This Directory contains a File with same Name

announcement.php

Existing File

3,940 bytes

Last changed 06/08/2003 00:00

Replace with

6,336 bytes

Last changed 13/03/2004 11:10

☑ All

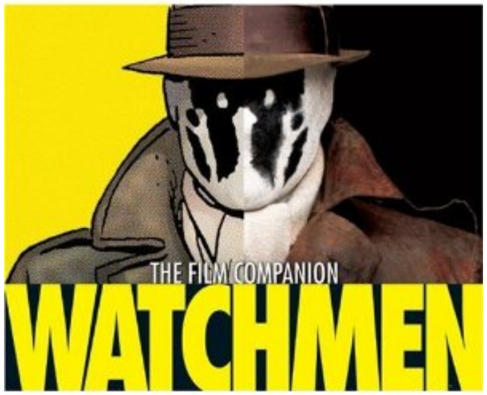
Insertion: Overwritten into File

- In this strategy, the virus would overwrite parts of the original executable
  - Advantage: The file size does not change
- Of course, overwriting could break the original infected file
- Possibilities
  - Overwrite repeated values, and restore them after execution
  - Move parts of file to innocent looking file (e.g., JPG)
  - Sometimes, executables are “padded” and there is unused space
  - Compress the original code, and decompress it later
- In any case, virus has to be small

Insertion: Companion Virus

- Companion virus
  - installs a COM file (the virus) for every EXE file found
  - idea is simple: DOS runs COM files before EXE
  - virus will stay memory resident and execute the original file
  - Question: What do you think of this infection strategy?

    → easy to find and eliminate



Insertion: NTFS ADS Viruses

- NTFS contains a system called Alternate Data Streams (ADS)
  - sometimes used by viruses
  - original intention of ADS is to store meta information with file e.g., has it been downloaded from the Internet?

```
echo ‘Hello World’ > test.txt
echo ‘This is Hidden’ > test.txt:hidden.txt
nodepad test.txt:hidden.txt
```

- Stream we have created is completely invisible
  - most commands do not work on ADSs (e.g., deleting).
  - Explorer and dir will not show the file
  - viruses can make use of ADS to hide code, data, temporary files
  - tool called *streams.exe* from Sysinternals.com is useful for finding such streams

NTFS ADS Demo

Insertion: Integration

- Code Integration
  - merge virus code with program
  - requires disassembly of target
    - difficult task on x86 machines
  - W95/Zmist is a classic example for this technique

Fast and Slow Infectors

- A fast infector infects any file accessed
  - purpose of fast infection is to ride on the back of anti-virus software
  - infect files as they are being checked
  - can be defeated if the scanner is started from a floppy

- A slow infector only infects files as they are created or modified
  - purpose of slow infection is to attempt to defeat integrity checking
  - piggyback on top of the process which legitimately changes a file
  - if integrity checker has a scanning component, virus can be caught

Tunneling and Camouflage Viruses

- To minimize the probability of its being discovered, a virus could use a number of different techniques

- A tunneling virus attempts to bypass antivirus programs
  - idea is to follow the interrupt chain back down to basic operating system or BIOS interrupt handlers
  - install virus there
  - virus is “underneath” everything – including the checking program
- In the past, possible for a virus to spoof a scanner by camouflaging itself to look like something the scanner was programmed to ignore
  - false alarms of scanners make “ignore” rules necessary

Sparse Infectors and Armored Viruses

- Sparse infector
  - infect every nth time a file is executed
  - infect files only with a certain name

- Armored virus
  - aims to make disassembly difficult
  - exploits fact that x86 code is hard to disassemble
  - Whale (early virus), made extensive use of such techniques
  - manual disassembly is almost always possible but takes more time and is not automated