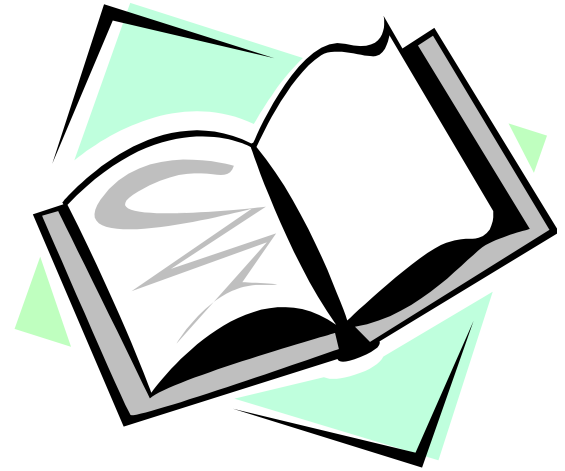# Information Security
CS 526
Topic 14

Malwares

Readings for This Lecture

- Wikipedia
  - Malware
  - Computer Virus
  - Botnet
  - Rootkit
  - Morris Worm

Malware Types

- Infectious:
  - Viruses, worms
- Concealment:
  - Trojan horses, and backdoors (trapdoors), logic bombs, rootkits
- Malware for stealing information:
  - Spyware, keyloggers, screen scrapers
- Malware for profit:
  - Dialers, scarewares, ransomware
- Botnets
- Many malwares have characterstics of multiple types

Trojan Horse

- Software that appears to perform a desirable function for the user prior to run or install, but (perhaps in addition to the expected function) steals information or harms the system.

- User 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
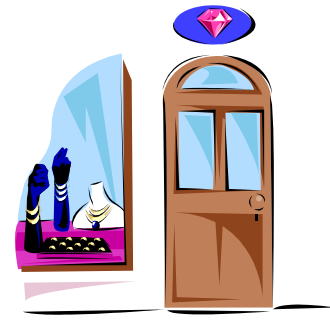
chmod u+s,o+x /tmp/.xxsh

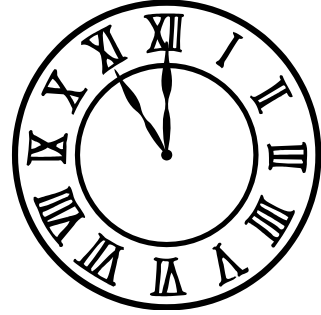**rm ./ls**

ls $*

as /homes/victim/ls

- *Victim*

ls

Trapdoor or Backdoor

- Secret entry point into a system
  - Specific user identifier or password that circumvents normal security procedures.

- Commonly used by developers
  - Could be included in a compiler.

Logic Bomb

- Embedded in legitimate programs
- Activated when specified conditions met
  - E.g., presence/absence of some file; Particular date/time or particular user
- When triggered, typically damages system
  - Modify/delete files/disks

Example of Logic Bomb

- In 1982, the Trans-Siberian Pipeline incident occurred. A KGB operative was to steal the plans for a sophisticated control system and its software from a Canadian firm, for use on their Siberian pipeline. The CIA was tipped off by documents in the Farewell Dossier and had the company insert a logic bomb in the program for sabotage purposes. This eventually resulted in "the most monumental non-nuclear explosion and fire ever seen from space".

Spyware

- Malware that collects little bits of information at a time about users without their knowledge
  - Keyloggers: stealthly tracking and logging key strokes
  - Screen scrapers: stealthly reading data from a computer display
  - May also tracking browsing habit
  - May also re-direct browsing and display ads

Scareware

- Software
  - with malicious payloads, or of limited or no benefit
  - Sold by social engineering to cause shock, anxiety, or the perception of a threat

- Rapidly increasing
  - Anti-Phishing Working Group: # of scareware packages rose from 2,850 to 9,287 in 2nd half of 2008.
  - In 1st half of 2009, the APWG identified a 583% increase in scareware programs.
  - A 2010 study by Google found 11,000 domains hosting fake anti-virus software, accounting for 50% of malware delivered via Internet advertising

SECURITY WARNING!
serious security threat detected

Your computer is infected with Spyware.
Your Security and Privacy are in DANGER.

Spyware programs can steal your credit card numbers and
bank information details. The computer can be used for sending
spam and you may get popups with adult or any other
unwanted content.

If
- You have visited adult or warez websites during past 3 days.
- Your homepage has changed and does not change back.
- Your computer performance has dropped down dramatically.
- You are suspecting someone is watching you.

Then your computer is most likely

INFECTED WITH SPYWARE.

We are sorry, but the trial version is
unable to remove these threats.
We strongly recommend you to purchase Full version.

You will get 24x7 friendly support and unlimited protection.

Continue Unprotected

Get Full version of SpySheriff Now!

Ransomware

- Holds a computer system, or the data it contains, hostage against its user by demanding a ransom.
  - Disable an essential system service or lock the display at system startup
  - Encrypt some of the user's personal files, originally referred to as **cryptoviruses**, **cryptotrojans** or **cryptoworms**

- Victim user has to
  - enter a code obtainable only after wiring payment to the attacker or sending an SMS message
  - buy a decryption or removal tool

Virus

- Attach itself to a host (often a program) and replicate itself

- Self-replicating code
  - Self-replicating Trojan horses
  - Alters normal code with "infected" version

- Operates when infected code executed

  If *spread condition* then

       For *target files*

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

  Perform malicious action

  Execute normal program

Worm

- Self-replicating malware that does not require a host program

- Propagates a fully working version of itself to other machines

- Carries a payload performing hidden tasks
  - Backdoors, spam relays, DDoS agents; …

- Phases
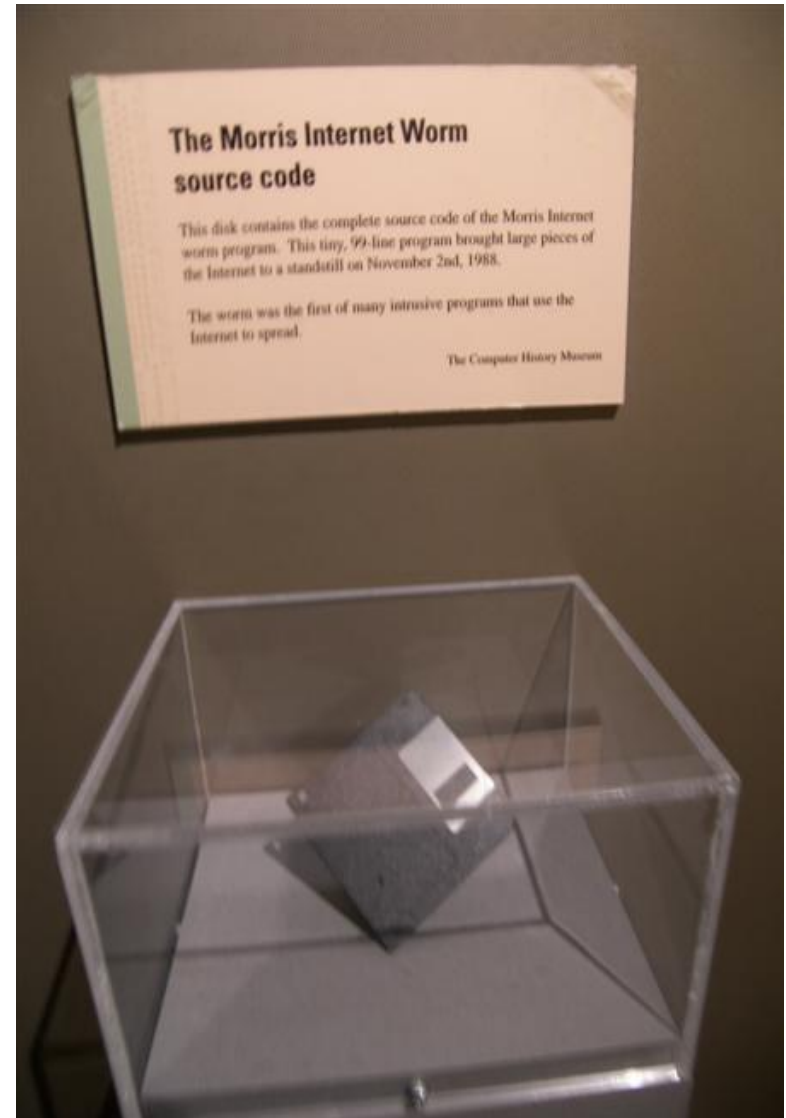 - Probing ➔ Exploitation ➔ Replication ➔ Payload

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

Morris Worm (November 1988)

- First major worm
- Written by Robert Morris
  - Son of former chief scientist of NSA's National Computer Security Center

The Morris Internet Worm source code

This disk contains the complete source code of the Morris Internet worm program. This tiny, 99-line program brought large pieces of the Internet to a standstill on November 2nd, 1988.

The worm was the first of many intrusive programs that use the Internet to spread.

The Computer History Museum

What comes next: *1 11 21 1211 111221?*

Morris Worm Description

- ## Two parts
  - ### Main program to spread worm
    - look for other machines that could be infected
    - try to find ways of infiltrating these machines

  - ### Vector program (99 lines of C)
    - compiled and run on the infected machines
    - transferred main program to continue attack

Vector 1: Debug feature of sendmail

- Sendmail
  - Listens on port 25 (SMTP port)
  - Some systems back then compiled it with DEBUG option on

- Debug feature gives
  - The ability to send a shell script and execute on the host

Vector 2: Exploiting fingerd

- What does finger do?
- Finger output

  arthur.cs.purdue.edu% finger ninghui

  Login name: ninghui   In real life: Ninghui Li

  Directory: /homes/ninghui   Shell: /bin/csh

  Since Sep 28 14:36:12 on pts/15 from csdhcp-120-173 (9 seconds idle)

  New mail received Tue Sep 28 14:36:04 2010;

   unread since Tue Sep 28 14:36:05 2010

  No Plan.

Vector 2: Exploiting fingerd

- Fingerd
  - Listen on port 79

- It uses the function gets
  - Fingerd expects an input string
  - Worm writes long string to internal 512-byte buffer

- Overrides return address to jump to shell code

Vector 3: Exploiting Trust in Remote Login

- Remote login on UNIX
  - rlogin, rsh

- Trusting mechanism
  - Trusted machines have the same user accounts
  - Users from trusted machines
  - /etc/host.equiv – system wide trusted hosts file
  - /.rhosts and ~/.rhosts – users' trusted hosts file

Host aaa.xyz.com
/etc/host.equiv
bbb.xyz.com

rlogin

Host bbb.xyz.com

User alice
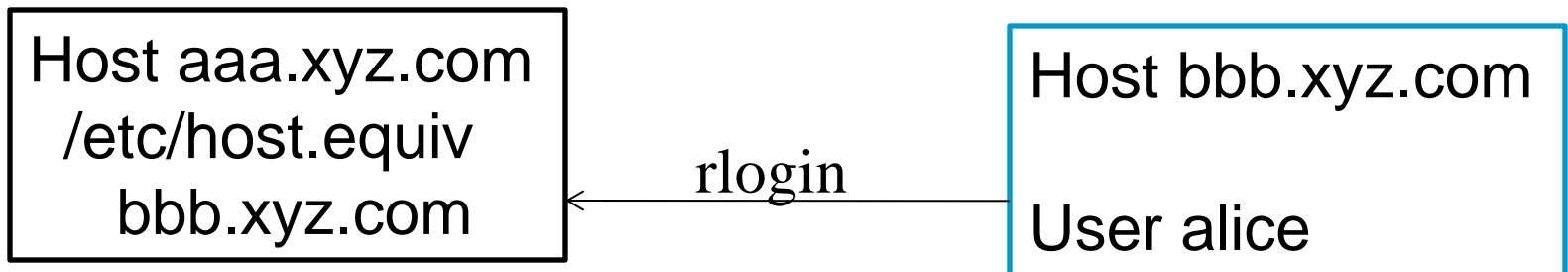
Vector 3: Exploiting Trust in Remote Login

- Worm exploited trust information
  - Examining trusted hosts files
  - Assume reciprocal trust
    - If X trusts Y, then maybe Y trusts X
- Password cracking
  - Worm coming in through fingerd was running as daemon (not root) so needed to break into accounts to use .rhosts feature
  - Read /etc/passwd, used ~400 common password strings & local dictionary to do a dictionary attack

Other Features of The Worm

- Self-hiding
  - Program is shown as 'sh' when ps
  - Files didn't show up in ls
- Find targets using several mechanisms:
  - 'netstat -r -n', /etc/hosts, …
- Compromise multiple hosts in parallel
  - When worm successfully connects, forks a child to continue the infection while the parent keeps trying new hosts
- Worm has no malicious payload
- Where does the damage come from?

Damage

- One host may be repeatedly compromised
- Supposedly designed to gauge the size of the Internet
- The following bug made it more damaging.
  - Asks a host whether it is compromised; however, even if it answers yes, still compromise it with probability 1/8.

Increasing propagation speed

- Code Red, July 2001
  - Affects Microsoft Index Server 2.0,
  - Exploits known buffer overflow in Idq.dll
  - Vulnerable population (360,000 servers) infected in 14 hours

- SQL Slammer, January 2003
  - Affects in Microsoft SQL 2000
  - Exploits known months ahead of worm outbreak
    - Buffer overflow vulnerability reported in June 2002
    - Patched released in July 2002 (Bulletin MS02-39)
  - Vulnerable population infected in less than 10 minutes

Slammer Worms (Jan., 2003)

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

SQLSERVR.EXE
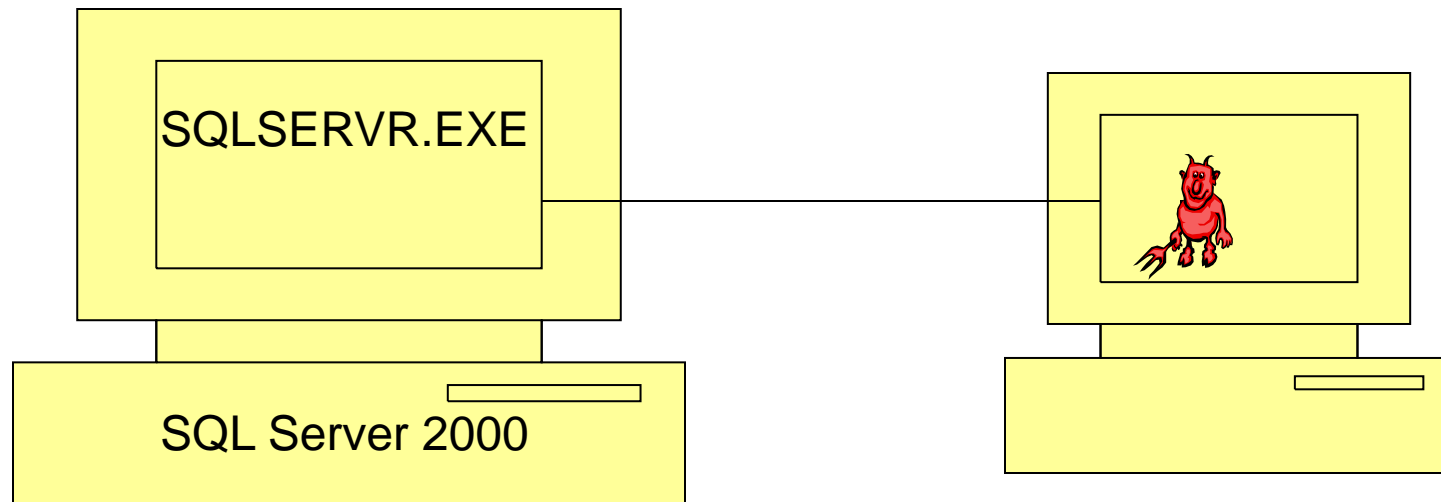
SQL Server 2000

Slammer's code is 376 bytes!

This byte signals the SQL Server to store the contents of the packet in the buffer

```
0000:  4500 0194 b6db 0000 6d11 c3c1 09e5 0a9c    E...¶Û..m.......
0010:  cb08 07c7                                   Ë..Ç.R....½² ....
       0101 0101                                   ...............
       0101 0101                                   ...............
       0101 0101                                   ...............
       0101 0101                                   ...............
0060:  0101 0101 0101 0101 0101 0101 0101 0101    ...............
0070:  0101 0101 0101 0101 0101 0101 01dc c9b0    ...............
0080:  42eb 0e01 0101 0101 0101 70ae 4201 70ae    Bë.......p.B.p.
0090:  4190 9090 9090 9090 9068 dcc9 b042 b301    B........hÜÉ°B³.
00a0:  0101 0131 c9b1 1850 e2fd 3501 0101 0550    ...1É±.Pâý5.....P
       2e64 6c6c 6865 6c33 3268 6b65             .äQh.dllhel32hke
       6f75 6e                                    rnQhounthic
                                                  tTf¹llQh32.
                                                  _f¹etQhsock
                               ff16               hsend¾..®B.
                               10ae               P.EàP.EðP..
                               10ae               B....=U.ìQt
                               049b               B...Ð1ÉQQP.
       0101 518d 45cc 508b 45c0 50ff             .ñ....Q.EÌ....
0140:  166a 116a 026a 02ff d050 8d45 c450 8b45    .j.j.j..ÐP.EÄP.E
0150:  c050 ff16 89c6 09db 81f3 3c61 d9ff 8b45    ÀP...Æ.Û..óa...E
0160:  b48d 0c40 8d14 88c1 e204 01c2 c1e2 0829    ´..@...Áâ..ÂÁâ.)
0170:  c28d 0490 01d8 8945 b46a 108d 45b0 5031    Â....Ø.E´j..E°P1
0180:  c951 6681 f178 0151 8d45 0350 8b45 ac50    ÉQf.ñx.Q.E.P.E¬P
0190:  ffd6 ebca                                   .ÖëÊ
```

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

value over
s and points it to a location
lsort.dll which effectively
calls a jump to %esp

Restore payload, set up socket structure, and get the seed for the random number generator

Research Worms

- Warhol Worms
  - infect all vulnerable hosts in 15 minutes – 1 hour
  - optimized scanning
    - initial hit list of potentially vulnerable hosts
    - local subnet scanning
    - permutation scanning for complete, self-coordinated coverage

- Flash Worms
  - infect all vulnerable hosts in 30 seconds
  - determine complete hit list of servers with relevant service open and include it with the worm

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

Nimda worm (September 18, 2001)

- Key Vulnerability to Exploit
  - **Microsoft Security Bulletin (MS01-020):** March 29, 2001
  - A logic bug in IE's rendering of HTML
  - Specially crafted HTML email can cause the launching of an embedded email

- Vector 1: e-mails itself as an attachment (every 10 days)
  - runs once viewed in preview plane

- Vector 2: copies itself to shared disk drives on networked PCs
  - Why this may lead to propagating to other hosts?

Nimda Worm

- Vector 3: Exploits various IIS directory traversal vulnerabilities
  - Use crafted URL to cause a command executing at
  - Example of a directory traversal attack:
    - http://address.of.iis5.system/scripts/..%c1%1c../winnt/system32/cmd.exe?/c+dir+c:\

- Vector 4: Exploit backdoors left by earlier worms

- Vector 5: Appends JavaScript code to Web pages

```
<script language="JavaScript">
window.open("readme.eml", null, "resizable=no,top=6000,left=6000")
</script>
```

Nimda worm

- 'Nimda fix' Trojan disguised as security bulletin
  - claims to be from SecurityFocus and TrendMicro
  - comes in file named FIX_NIMDA.exe
    - TrendMicro calls their free Nimda removal tool FIX_NIMDA.com

Storm botnet

- First detected in Jan 2007
- Vectors (primarily social engineering):
  - Email attachments
  - Download program to show a video
  - Drive-by exploits
- DDoS spam fighting sites, and whichever host discovered to investigate the botnet
- Peer-to-peer communications among bots
  - for asking for C&C server

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

Antivirus

- Typical strategies
  - Check for integrity of known good files
  - Check for known malware, using signatures,
    - Each signature is a sequence of bytes that an antivirus program looks for because it is known to be part of the virus
  - Main challenges
    - Reduce false positives/false negatives
    - Improve speed of checking

Malware Anti-detection Strategy

- Encrypt malware code using variable key
  - Self modifying code may look suspicious
  - Decryption part may become signature
- Polymorphic code
  - Use encryption, but also change decryption routine each time
- Metamorphic
  - Use an engine each time to generate a functional equivalent but different code

Rootkit

- A **rootkit** is software that enables continued privileged access to a computer while actively hiding its presence from administrators by subverting standard operating system functionality or other applications.

- Emphasis is on hiding information from administrators' view, so that malware is not detected
  – E.g., hiding processes, files, opened network connections, etc

- Example: Sony BMG copy protection rootkit scandal
  – In 2005, Sony BMG included Extended Copy Protection on music CDs, which are automatically installed on Windows on CDs are played.

Types of Rootkits

- ## User-level rootkits
  - Replace utilities such as ps, ls, ifconfig, etc
  - Replace key libraries
  - Detectable by utilities like tripwire

- ## Kernel-level rootkits
  - Replace or hook key kernel functions
  - Through, e.g., loadable kernel modules or direct kernel memory access
  - A common detection strategy: compare the view obtained by enumerating kernel data structures with that obtained by the API interface
  - Can be defended by kernel-driver signing (required by 64-bit windows)

More Rootkits

- Bootkit (variant of kernel-level rootkit)
  - Replace the boot loader (master boot record)
  - Used to attack full disk encryption key
  - Malicious boot loader can intercept encryption keys or disable requirement for kernel-driver signing

- Hypervisor-level rootkits
- Hardware/firmware rootkits
- Whoever gets to the lower level has the upper hand.

How does a computer get infected with malware or being intruded?

- Executes malicious code via user actions (email attachment, download and execute trojan horses, or inserting USB drives)

- Buggy programs accept malicious input
  - daemon programs that receive network traffic
  - client programs (e.g., web browser, mail client) that receive input data from network
  - Programs Read malicious files with buggy file reader program

- Configuration errors (e.g., weak passwords, guest accounts, DEBUG options, etc)

- Physical access to computer

Coming Attractions …

- Software Security