Network Security

Intruders and Viruses

Password Management Part Two - Cracking

Intrusion Techniques

- Objective: Gain access to a system
- Frequent Goal: Acquiring a user password
- Most systems have a file that maps a password to each user
- Password file protection:
  - one-way encryption
  - access control

Password Learning Techniques

guess attack

1. Try default passwords used with standard accounts shipped with the system
2. Exhaustive try of all short passwords
3. Try words in system’s dictionary or list of likely passwords (hacker bulletin boards)
4. Collect information about users (full names, names of spouses and children, pictures and books in their office, related hobbies)
5. Try users’ phone numbers, social security numbers, room numbers
6. Try all legitimate license plate numbers
7. Use a trojan horse
8. Tap the line between a remote user and the system

Password Protection

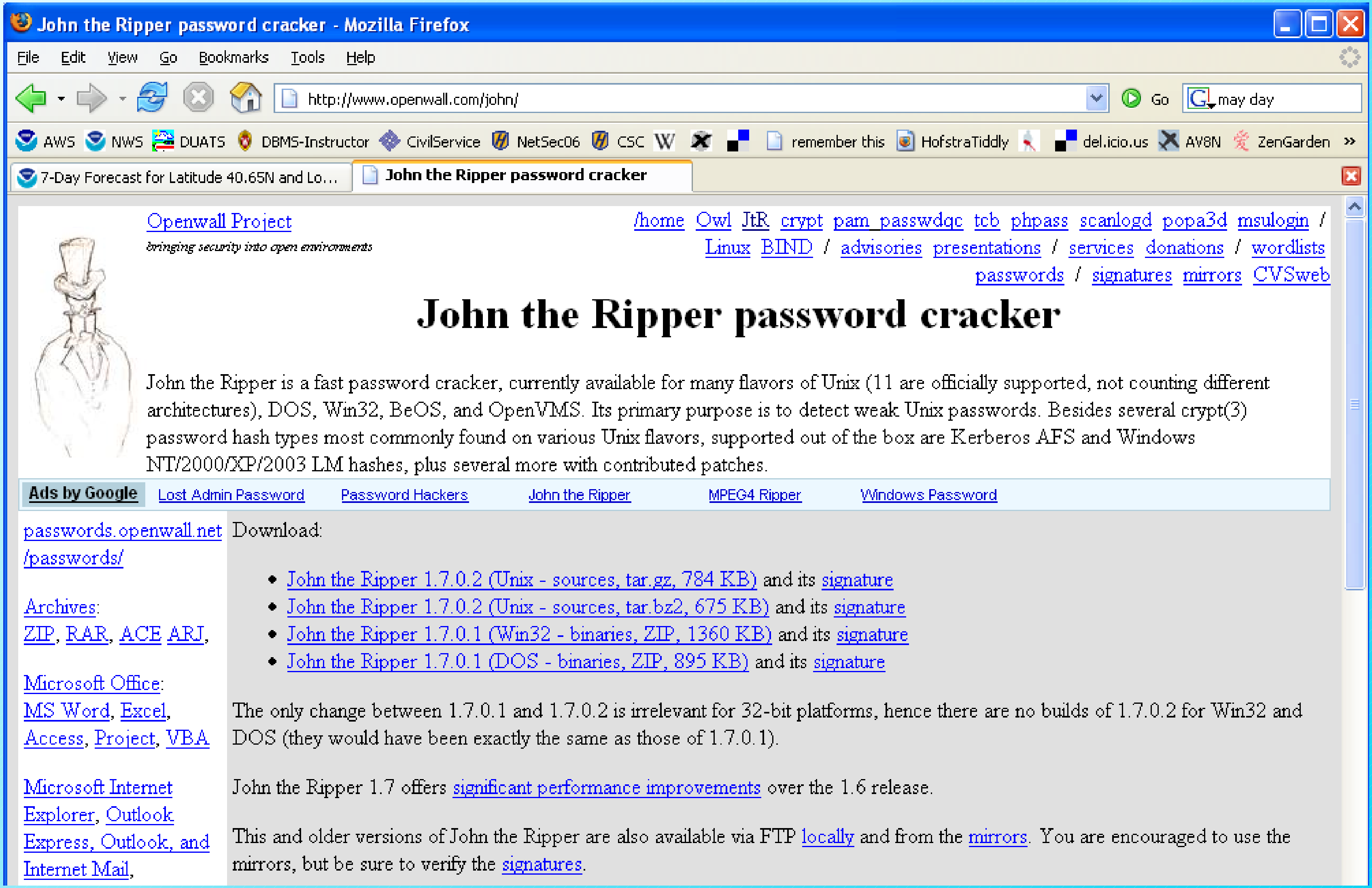
Unix password scheme threats:

- Gain access through a guest account and run a password cracker
- Obtain a copy of the password file and run a password cracker

Goal: Run a password cracker

- Rely on people choosing easily guessable passwords!

Password Cracking


John the Ripper password cracker - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://www.openwall.com/john/
Go
may day
AWS NWS DUATS DBMS-Instructor CivilService NetSec06 CSC remember this HofstraTiddly del.icio.us AV8N ZenGarden »
7-Day Forecast for Latitude 40.65N and Lo...
John the Ripper password cracker

Openwall Project

bringing security into open environments

/home Owl JtR crypt pam_passwdqc tcb phpass scanlogd popa3d msulogin / Linux BIND / advisories presentations / services donations / wordlists passwords / signatures mirrors CVSweb

John the Ripper password cracker

John the Ripper is a fast password cracker, currently available for many flavors of Unix (11 are officially supported, not counting different architectures), DOS, Win32, BeOS, and OpenVMS. Its primary purpose is to detect weak Unix passwords. Besides several crypt(3) password hash types most commonly found on various Unix flavors, supported out of the box are Kerberos AFS and Windows NT/2000/XP/2003 LM hashes, plus several more with contributed patches.

passwords.openwall.net /passwords/

Archives: ZIP, RAR, ACE ARJ,

Microsoft Office: MS Word, Excel, Access, Project, VBA

Microsoft Internet Explorer, Outlook Express, Outlook, and Internet Mail,

Download:

- John the Ripper 1.7.0.2 (Unix - sources, tar.gz, 784 KB) and its signature
- John the Ripper 1.7.0.2 (Unix - sources, tar.bz2, 675 KB) and its signature
- John the Ripper 1.7.0.1 (Win32 - binaries, ZIP, 1360 KB) and its signature
- John the Ripper 1.7.0.1 (DOS - binaries, ZIP, 895 KB) and its signature

The only change between 1.7.0.1 and 1.7.0.2 is irrelevant for 32-bit platforms, hence there are no builds of 1.7.0.2 for Win32 and DOS (they would have been exactly the same as those of 1.7.0.1).

John the Ripper 1.7 offers significant performance improvements over the 1.6 release.

This and older versions of John the Ripper are also available via FTP locally and from the mirrors. You are encouraged to use the mirrors, but be sure to verify the signatures.

Password Cracking

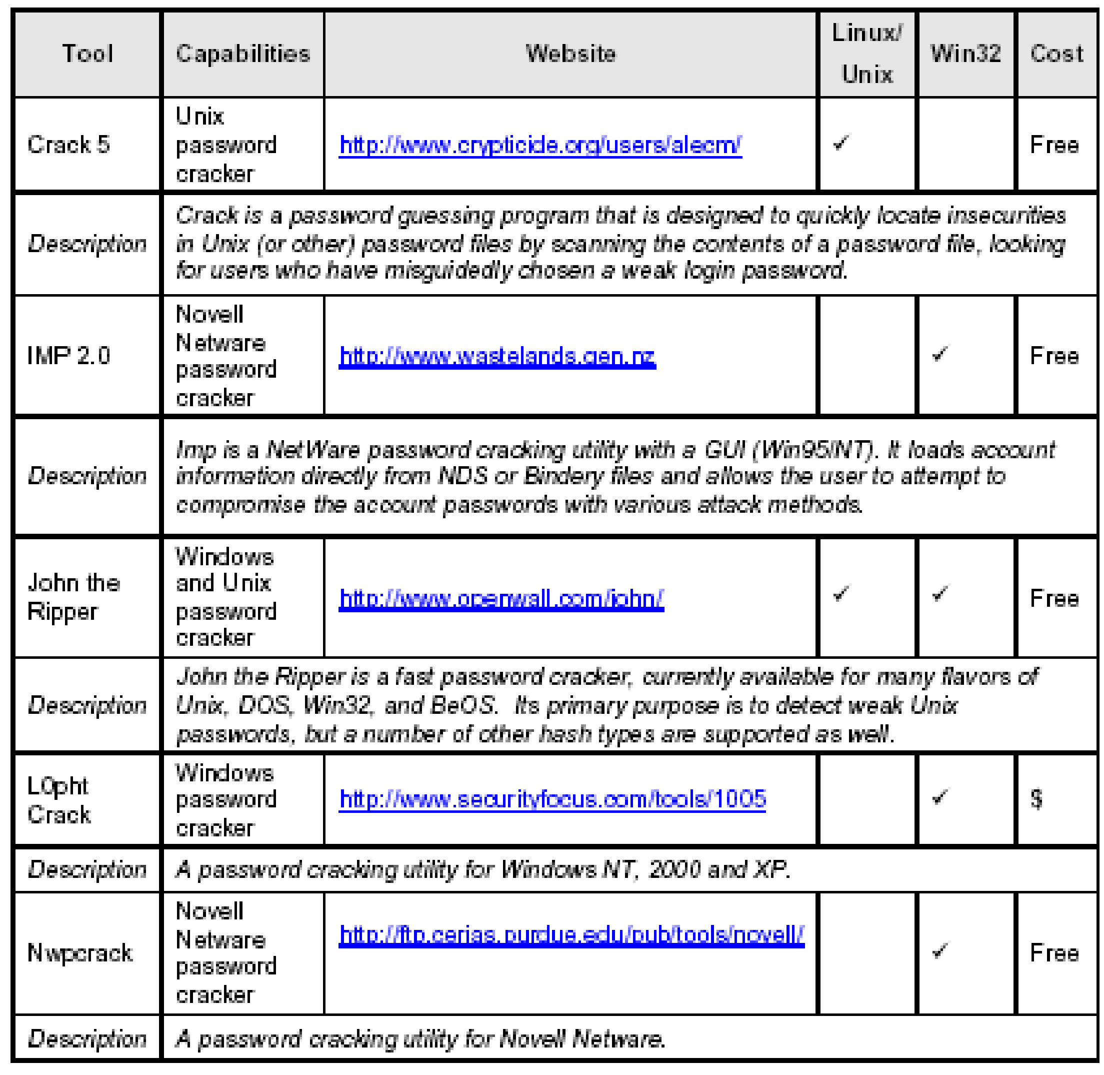
Unix Password File (/etc/passwd):

```
daemon:x:1:1::/:
bin:x:2:2::/usr/bin:
sys:x:3:3::/:
nobody:x:60001:60001:Nobody:/:
eric:GmTFg0AavFA0U:1001:10:Eric Schwartz:/export/home/eric:/bin/ksh
temp:kRWegG5iTZP5o:1002:10:IP Administration:/export/home/ipadmin:/bin/ksh
jfr:kyzKROryhFDE2:506:506::/home/jfr:/bin/csh
```

Results of the password cracker:

```
$ john  passwd
Loaded 3 passwords with 3 different salts (Standard DES [24/32 4K])
temp             (temp)
jenny            (eric)
solaris1         (jfr)
```

Password Crackers



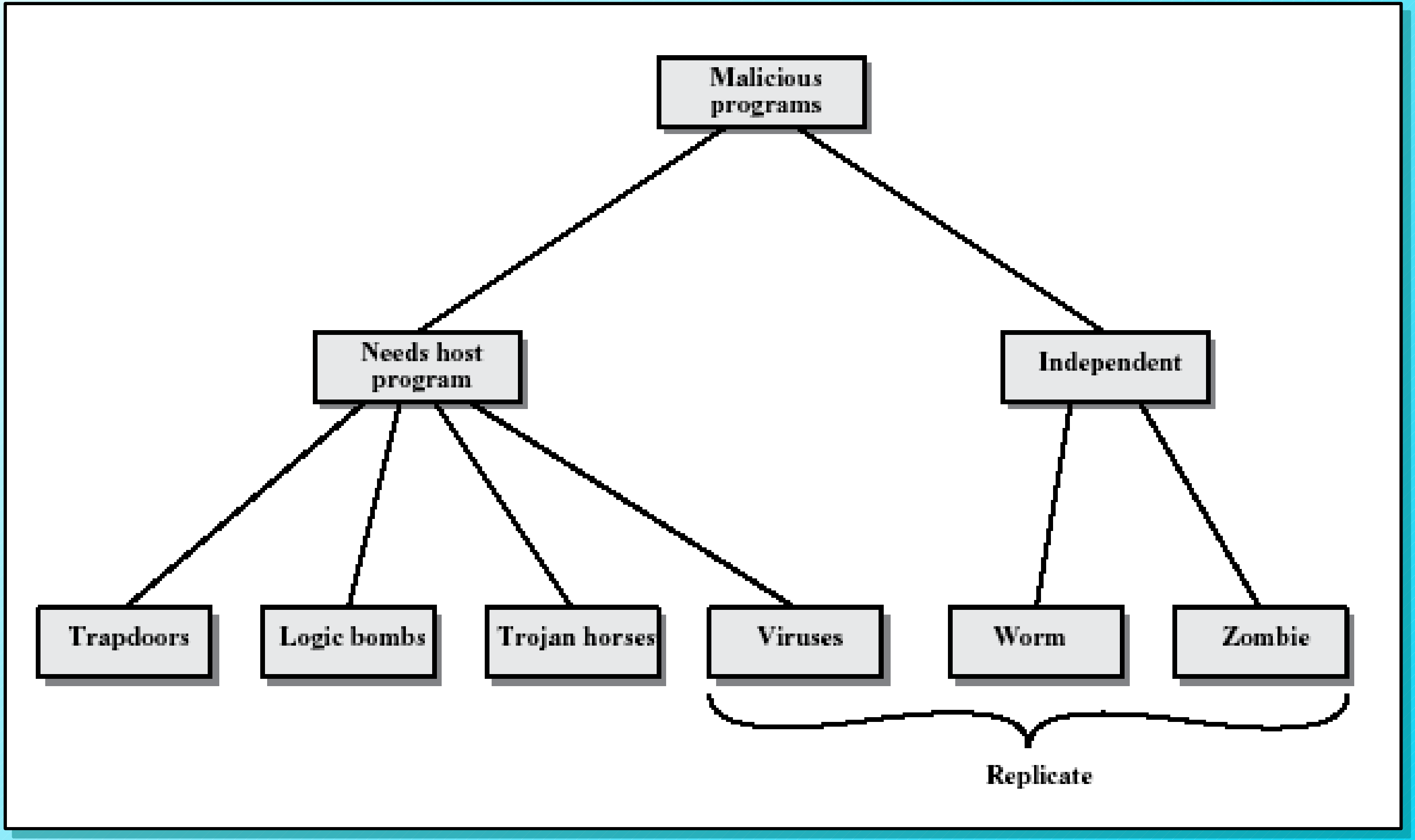
Tool	Capabilities	Website	Linux/ Unix	Win32	Cost
Crack 5	Unix password cracker	http://www.crypticide.org/users/alecm/	✓		Free
Description	*Crack is a password guessing program that is designed to quickly locate insecurities in Unix (or other) password files by scanning the contents of a password file, looking for users who have misguidedly chosen a weak login password.*				
IMP 2.0	Novell Netware password cracker	http://www.wastelands.gen.nz		✓	Free
Description	*Imp is a NetWare password cracking utility with a GUI (Win95/NT). It loads account information directly from NDS or Bindery files and allows the user to attempt to compromise the account passwords with various attack methods.*				
John the Ripper	Windows and Unix password cracker	http://www.openwall.com/john/	✓	✓	Free
Description	*John the Ripper is a fast password cracker, currently available for many flavors of Unix, DOS, Win32, and BeOS. Its primary purpose is to detect weak Unix passwords, but a number of other hash types are supported as well.*				
L0pht Crack	Windows password cracker	http://www.securityfocus.com/tools/1005		✓	$
Description	*A password cracking utility for Windows NT, 2000 and XP.*				
Nwpcrack	Novell Netware password cracker	http://ftp.cerias.purdue.edu/pub/tools/novell/		✓	Free
Description	*A password cracking utility for Novell Netware.*				

Virus and Related Threats

Malicious Programs

- Two categories:
  - Those that need a host program – fragments of programs - parasitic
  - Those that are independent – self contained
- Some replicate – used as a differentiator

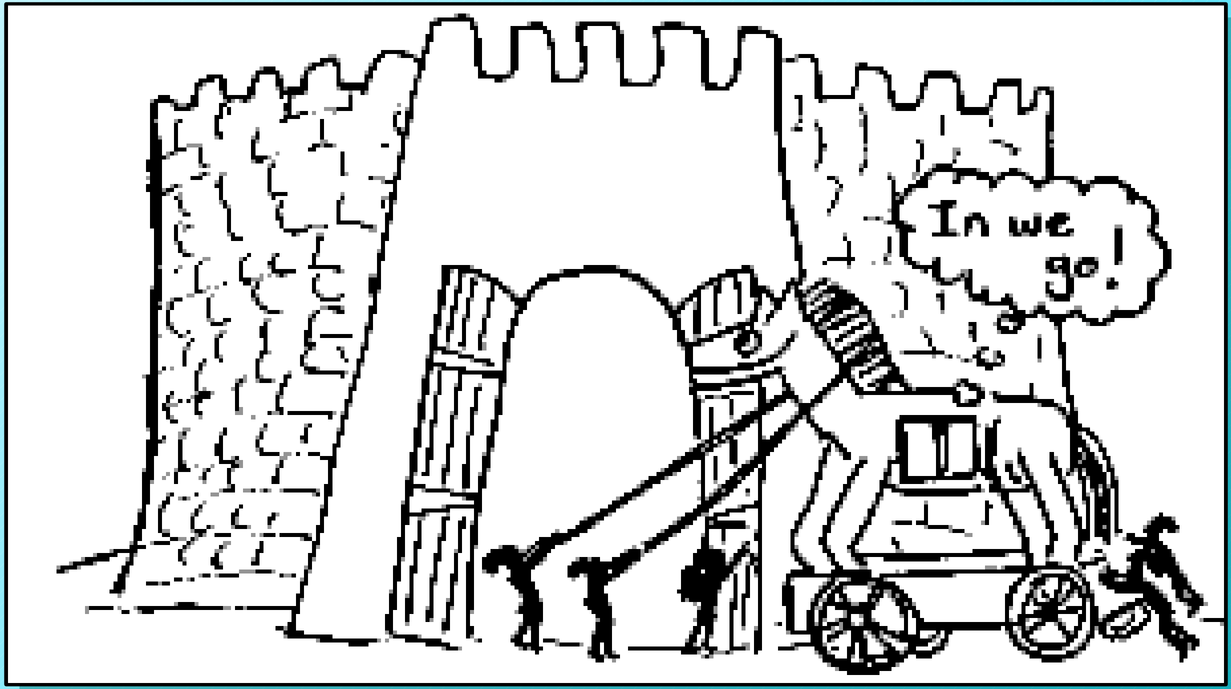
Taxonomy of Malicious Programs



Malicious Programs

- Logic Bombs: logic embedded in a program that checks for a set of conditions to arise and executes some function resulting in unauthorized actions
- Trapdoors: secret undocumented entry point into a program, used to grant access without normal methods of access authentication (*e.g.,War Games*)

Trojan Horse



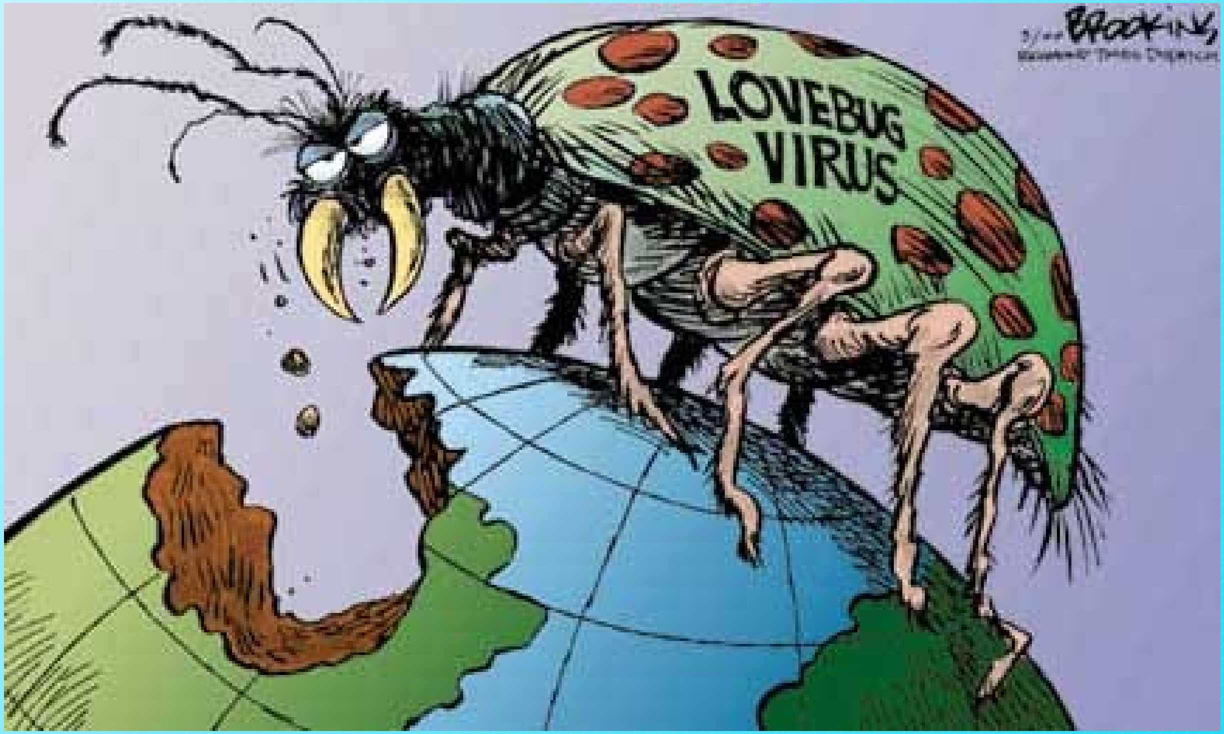
Malicious Programs

- Trojan Horse: secret undocumented routine embedded within a useful program, execution of the program results in execution of the routine
- Common motivation is data destruction

Malicious Programs

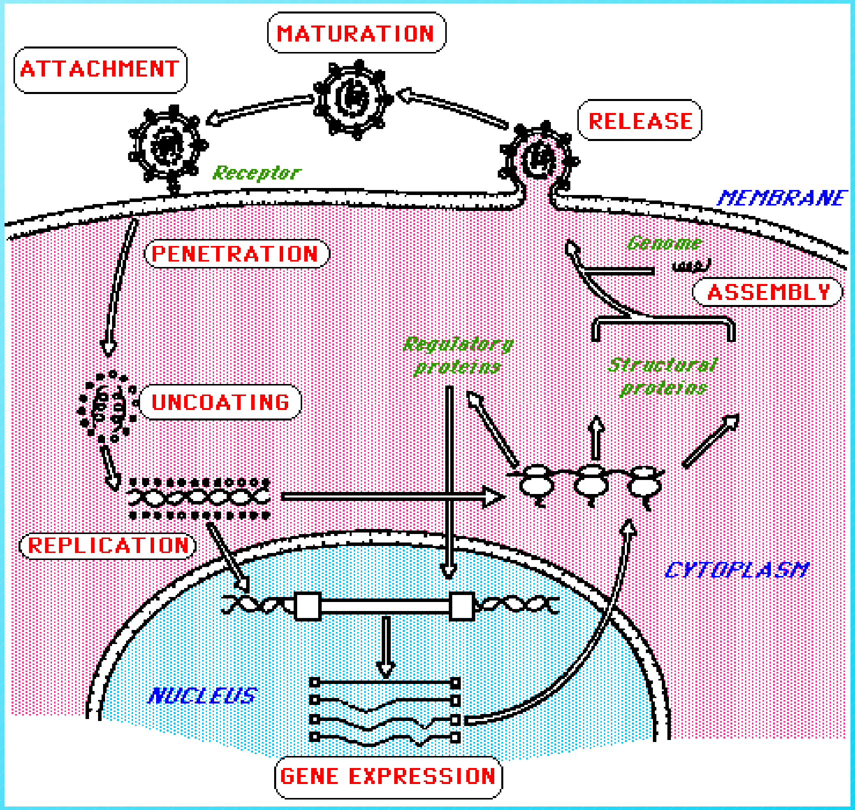
- Zombie: a program that secretly takes over an Internet attached computer and then uses it to launch an untraceable attack
- Very common in Distributed Denial-Of-Service attacks

Viruses



Viruses

- A virus is a submicroscopic parasitic particle that infects cells in biological organisms.
- Viruses are non-living particles that can only replicate when an organism reproduces the viral RNA or DNA.
- Viruses are considered non-living by the majority of virologists
- www.virology.net



Viruses

- Viruses: code embedded within a program that causes a copy of itself to be inserted in other programs and performs some unwanted function
- *Infects* other programs
- *Code* is the *DNA* of the virus

Worms

"Computer Worm" Copyright John S. Pritchett

Worms

- Worms: program that can replicate itself and send copies to computers across the network and performs some unwanted function
- Uses *network connections* to spread from system to system

Bacteria

- Bacteria: *consume resources* by replicating themselves
- Do not explicitly damage any files
- *Sole purpose* is to *replicate* themselves
- Reproduce exponentially
- Eventually taking up all processors, memory or disk space

Nature of Viruses

Four stages of virus lifetime

- Dormant phase: virus idle
- Propagation phase: cloning of virus
- Triggering phase: virus activation
- Execution phase: unwanted function performed

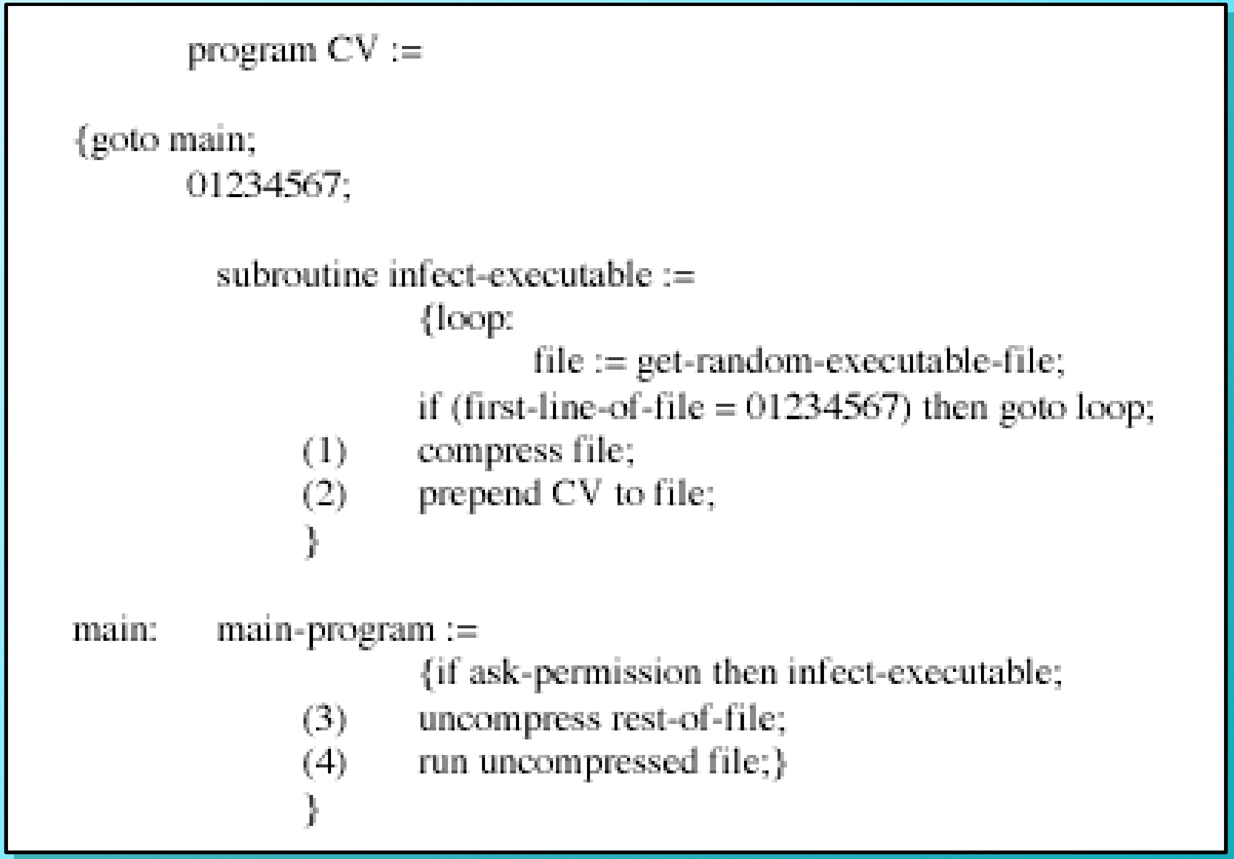
Virus Structure

```
program V:=

{goto main:
    1234567;    ←   special marker determines if infected

       subroutine infect-executable :=
           {loop:
           file:= get-random-executable-file;
           if (first-line-of-file = 1234567)
           then goto loop
           else prepend V to file;}

       subroutine do–damage :=
           {whatever damage is to be done}

       subroutine trigger-pulled :=
           {return true if some condition holds}

main:  main-program :=
       {infect-executable;
       if trigger-pulled then do-damage;
       goto next;}
next:  ←   transfer control to the original program

}
```

Avoiding Detection

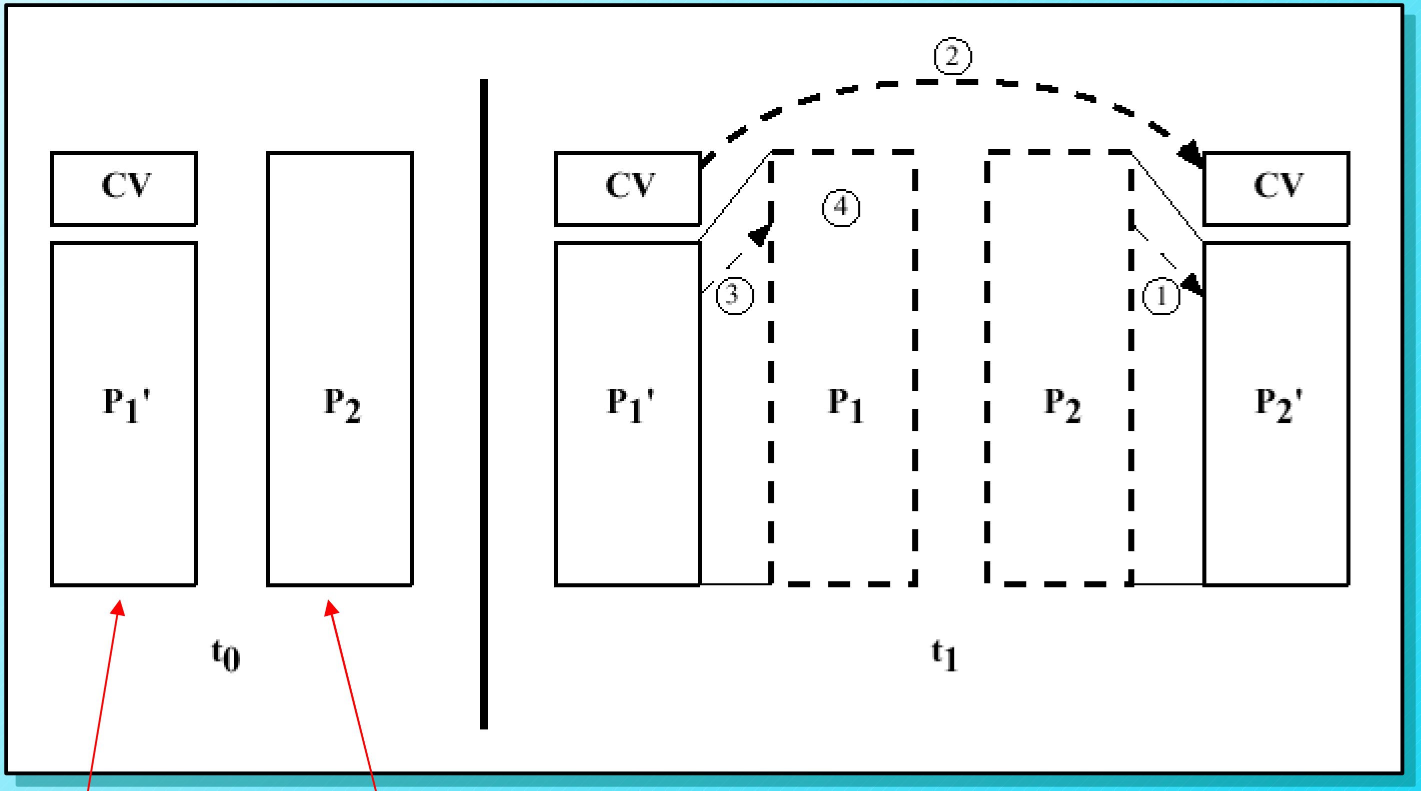
- Infected version of program is longer than the corresponding uninfected one
- *Solution:* compress the executable file so infected and uninfected versions are identical in length

Avoiding Detection



```
          program CV :=

{goto main;
          01234567;

            subroutine infect-executable :=
                          {loop:
                                  file := get-random-executable-file;
                          if (first-line-of-file = 01234567) then goto loop;
               (1)        compress file;
               (2)        prepend CV to file;
               }

main:       main-program :=
                          {if ask-permission then infect-executable;
               (3)        uncompress rest-of-file;
               (4)        run uncompressed file;}
               }
```

Compression Program



infected uninfected

Types of Viruses

- Parasitic Virus: attached to executables, replicates when program is executed
- Memory-resident virus: part of a resident system program, affects every program executed
- Boot sector virus: infects a master boot record and spreads when system is booted from infected disk

Types of Viruses

- Stealth virus: virus designed to hide itself from detection by antivirus software (compression, interception of I/O logic)
- Polymorphic virus: mutates with every infection making detection by “signature” impossible (mutation engine)
- Macro virus: infects Microsoft Word docs; 2/3’s of all viruses

Macro Viruses

- 2/3s of all viruses
- Mainly Microsoft products – platform independent
- Affect documents not executables
- Easily spread by e-mail
- Autoexecuting macro is the culprit

Worms

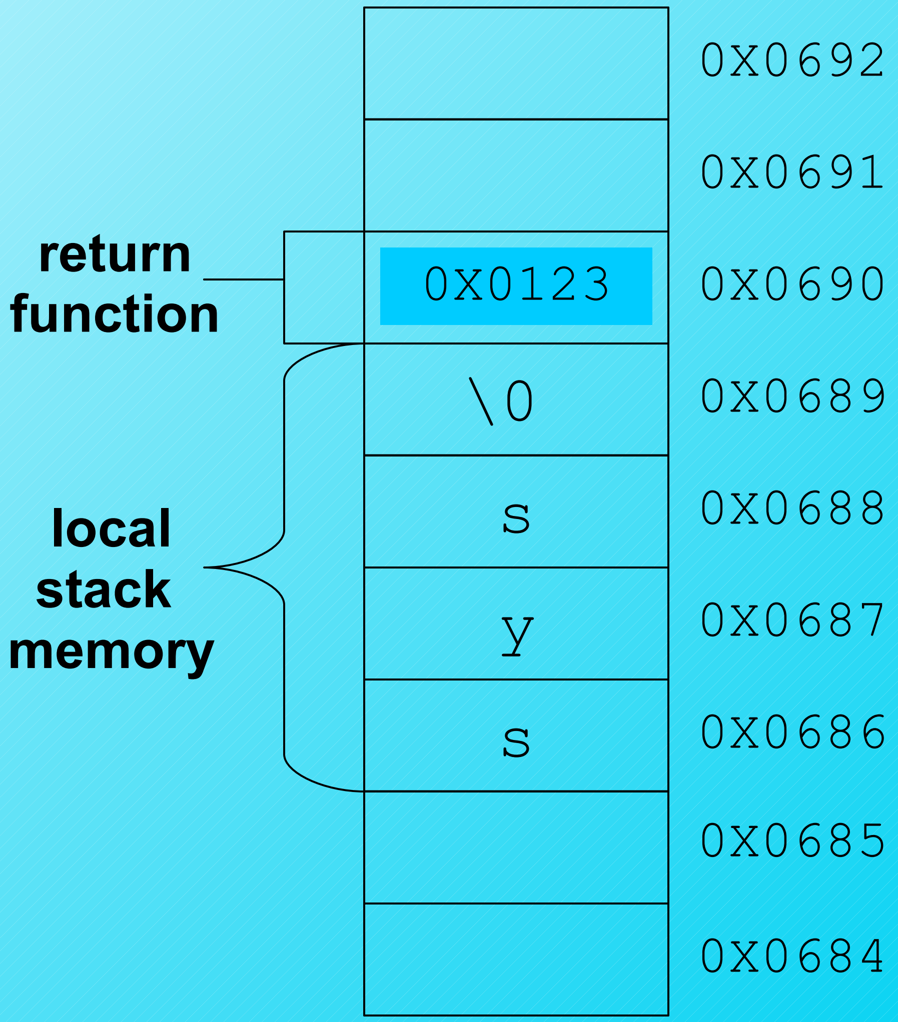
- Uses network connections to spread from system to system
- Similar to a virus – has same phases: dormant, propagation, trigger and execution
- Morris Worm – most famous
- Recent: OSX.Leap.A, Kama Sutra,Code Red

Buffer Overflow

- Program attempts to write more data into buffer than that buffer can hold…
- …Starts overwriting area of stack memory
- Can be used maliciously to cause a program to execute code of attackers choose
- Overwrites stack point

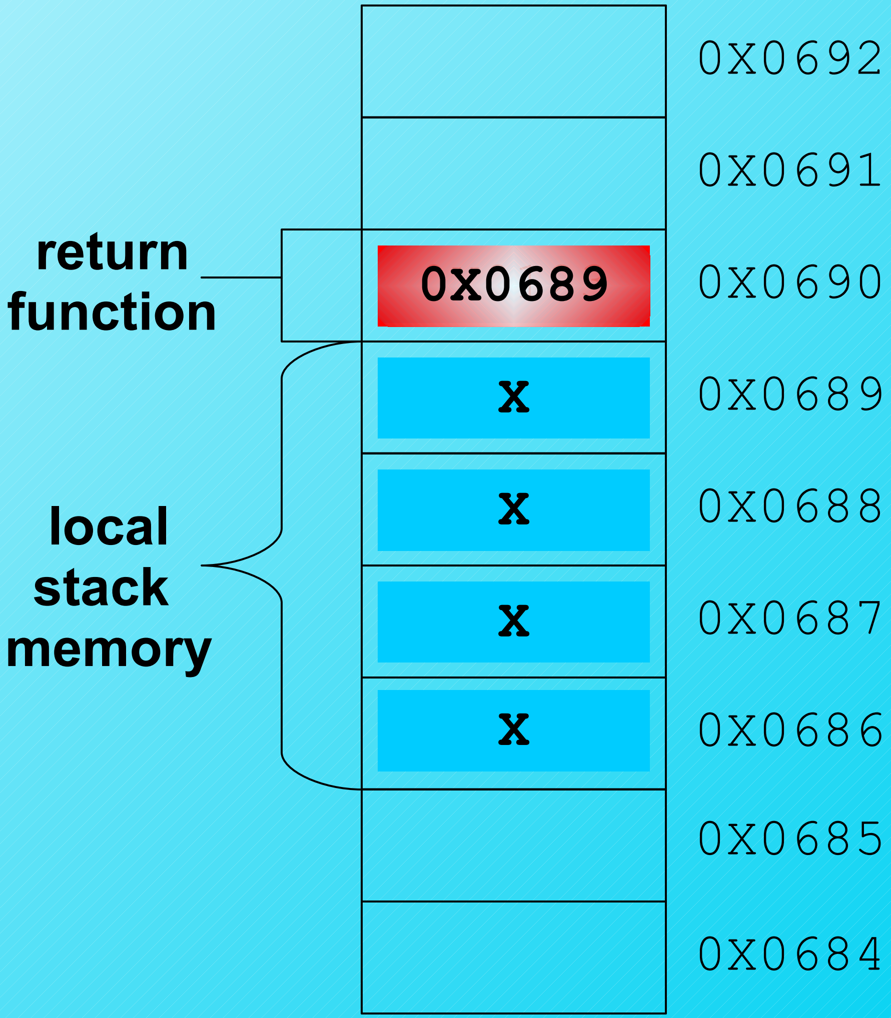
Mechanics of stack-based buffer overflow

- Stack is like a pile of plates
- When a function is called, the return address is pushed on the stack
- In a function, local variables are written on the stack
- Memory is written on stack
  - char username[4] reserved 4 bytes of space on stack



Mechanics of stack-based buffer overflow

- When function copies too much on the stack...
- ...the return pointer is overwritten
- Execution path of function changed when function ends
- Local stack memory has malicious code



Antivirus Approaches

- Detection – determine that it has occurred and locate the virus
- Identification – identify the specific virus
- Removal – remove all traces and restore the program to its original state

Generations of Antivirus Software

- First: simple scanners (record of program lengths)
- Second: heuristic scanners (integrity checking with checksums)
- Third: activity traps (memory resident, detect infected actions)
- Fourth: full-featured protection (suite of antivirus techniques, access control capability)

Advanced Techniques

- Generic Decryption
- Digital Immune System
- Behavior-Blocking Software

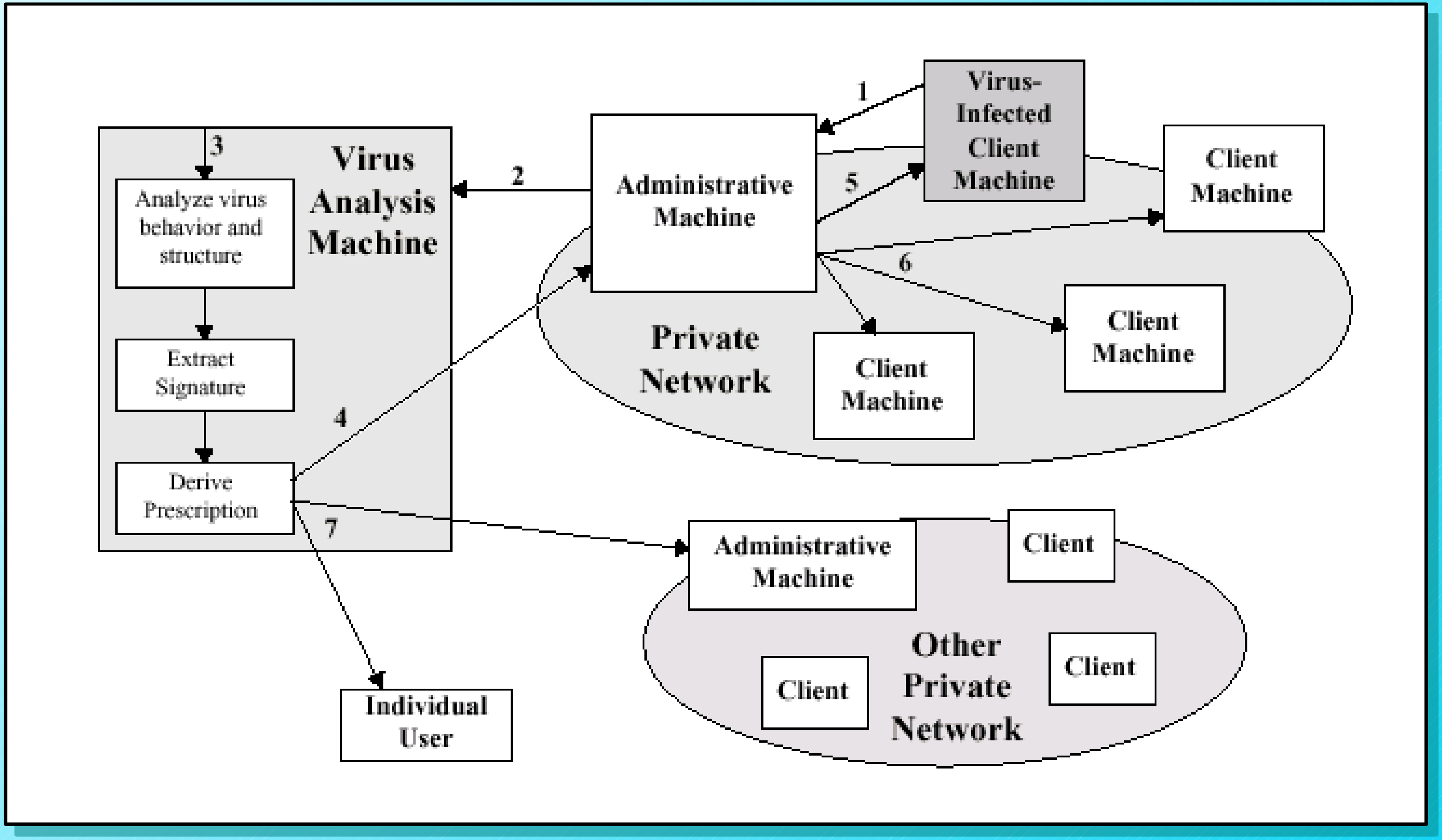
Generic Decryption

- Easily detects even most complex polymorphic virus
- No damage to the personal computer
- Contains following elements:
  - CPU emulator – software based virtual computer
  - Virus signature scanner – scans target code for known signatures
  - Emulation control module – control execution of target code

Digital Immune System

- Pioneered by IBM
- Response to rate of virus propagation
  - Integrated mail systems - Outlook
  - Mobile program systems – ActiveX, Java
- Expands the use of program emulation
- Depends on a central virus analysis machines

Digital Immune System



Behavior-Blocking Software

- Monitors program behavior in real-time for malicious actions – part of OS
- Look for well defined requests to the OS: modifications to files, disk formats, mods to scripts or macros, changes in config settings, open network connections, etc.
- IPS – Intrusion Prevention Systems

Malicious Code Protection Types of Products

- Scanners - identify known malicious code - search for *signature strings*
- Integrity Checkers – determine if code has been altered or changed – *checksum* based
- Vulnerability Monitors - prevent modification or access to particularly sensitive parts of the system – user defined
- Behavior Blockers - list of rules that a legitimate program must follow – *sandbox* concept

Important URLs

- http://www.cert.org/ Originally DARPA’s computer emergency response team. An essential security site
- http://www.research.ibm.com/antivirus/ IBM’s site on virus information. Very good papers – a little outdated
- http://www.afsa.org/fsj/sept00/Denning.cfm Hacktivism: An Emerging Threat to Diplomacy, another Denning term along with Information Warfare
- http://csrc.nist.gov/virus/Computer Security Resources Center – Virus information and *alerts*

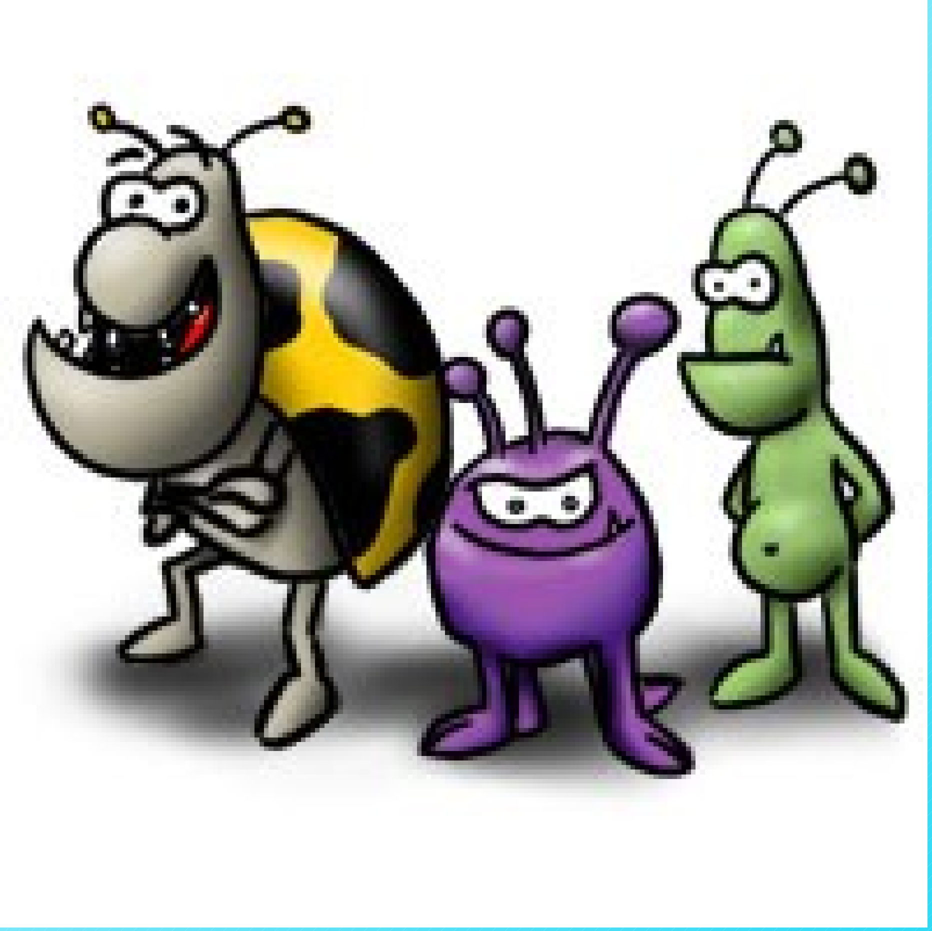
Important URLs

- http://www.ciac.org/ciac/
  Computer Incident Advisory Capability -another bookmark-able site to visit regularly
- http://csrc.nist.gov/publications/nistpubs/800-42/

  Guideline on Network Security Testing – covers password cracking
- http://www.openwall.com/john/
  Very good password cracker, “John the Ripper”
- http://csrc.nist.gov/publications/nistpubs/800-36/

  Guide to Selecting Information Security Products
- http://www.xensource.com/
  Xen Source - Hottest Area In Virtualization

… enough!



…coming to the end!

- Take Home Final Exam – On Website
- Due Next Class
- Return Papers
- Any Problems, Please Email Or Call
- Good Luck