

Virus Detection & Prevention

Current Techniques For Detecting and Preventing Damage From Computer Viruses

COMP 5370/6370



COMP 6370 – Virus Detection & Prevention



3 Basic Anti-Virus Technologies

- **3 Basic Anti-Virus Technologies**
 - **Virus Scanners**
 - **Integrity Checkers**
 - **Behavior Blockers**



Virus Scanners

- **Virus Scanners**

- Examine the contents of each file that can carry executable instructions
 - “.exe”, “.bat”, “.com”, “.vbs”, “.scr”, etc.
- Search each potential file for certain “search strings” which are present in known viruses.
- Use a variety of techniques to check for matches
 - **Fuzzy search (Heuristic search), exact search**
 - Fuzzy search accounts for virus variants by not requiring an exact match, takes more time
 - Exact search will not catch virus variants, but is much faster



Virus Scanning

- Search each file for a known “search string”
- Search string should be something that uniquely identifies the virus
 - In this case, searching for the two printf statements and the system() call may make a good signature
 - Remember, searching is parsing the hexadecimal executable, so there is no need to worry about case sensitivity, overhead of comparing strings, etc.

```
//COMP 6370 Virus
#include <stdio.h>
#include <stdlib.h>
{
printf(“The COMP 6370 virus\n”);
printf(“Removing C:\My Documents\n”);
system(“deltree C:\My Documents”);
}
```



Virus Scanners

- **Problems With Virus Scanners**
 - **Unable to cope with unknown viruses**
 - **Since scanners use a database of known viruses, unknown viruses will escape detection**
 - **Minor variants of known viruses can be missed**
 - **Fuzzy search is very time intensive, so software developers may not use it as aggressively as it should**
 - **Time concerns**
 - **Time required to compare the contents of each executable to each virus in the database can be very large**



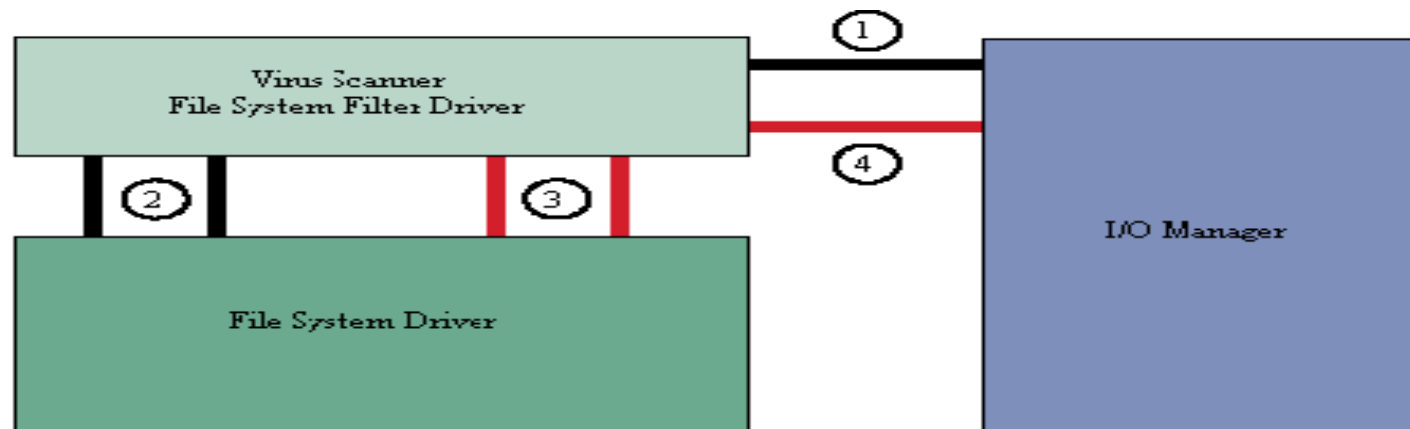
Virus Scanners

- **Solutions to time problem for virus scanners**
 - Many commercial virus scanners now use real-time scanning to speed up the process
 - Scan each file as it's opened or executed, instead of scanning them all at once
 - **New Problem:**
 - By examining each file as it's executed, the overall performance of the executable is decreased



Real-time (“On-Access”) Virus Scanner

- Model of real time virus scanner (aka “On-Access Scanner”)



- ① Original CreateFile IRP from I/O Manager
- ② CreateFile IRP propagated to file system
- ③ Virus scanner reads and checks file; if virus detected, virus scanner closes file
- ④ Virus scanner returns original CreateFile result, or failure if file contains virus



Integrity Checkers

- **Integrity Checkers**
 - Creates a checksum for each executable file in a directory, and stores the results in a file.
 - Each time the Integrity Checker is run, it re-computes the checksum for each executable file, and compares this value to the previously stored checksum.
 - If the values match, then the file is assumed to be clean.
 - If values do not match, executable has probably been infected by a virus.



Integrity Checkers

- **Problems With Integrity Checking**
 - **Virus can modify checksum file**
 - Integrity checker will compare computed checksum with checksum stored in the file, and will ignore the file
 - **Virus can delete the checksum file**
 - With the checksum file deleted, there is no basis for determining previous checksums
 - **Virus can encrypt checksum file**
 - Same problems as deleting the checksum file
 - **Integrity Checking only works for file infecting viruses**
 - Viruses that copy themselves to the hard disk (as many viruses do) will be ignored, since there is no checksum discrepancy



Behavior Blocking

- **Behavior Blocking**
 - Does not proactively search for known viruses
 - Instead monitors system for suspicious activity
 - **Example:**
 - Program *virus.exe* suddenly attempts to delete all .mp3 files stored on the hard disk
 - Program *word.exe* attempts to delete all .jpg files from the “My Documents” folder
 - If suspicious activity is observed, consult a list of rules to determine appropriate action
 - **Program is allowed to continue, and performs desired operation**
 - **Program is terminated before attempted operations are performed**
 - If no appropriate rule is found the user/administrator is consulted



Behavior Blocking

- **Behavior Blocking**
 - **Advantages:**
 - **More resistant to unknown threats**
 - **No new virus definitions to download – system does not necessarily require continual maintenance**
 - **Disadvantages:**
 - **Continuous monitoring of every aspect of system can greatly reduce system speed**
 - **Monitoring memory allocation, network access, file system access simultaneously is an expensive proposition**
 - **Many possible false positives**
 - **AI simply not evolved enough to correctly interpret every system action**
 - **System is not “bullet proof”**
 - **New viruses may be able to perform actions that do not get flagged, but can still be used to execute payload**
 - **New viruses may be able to emulate other programs installed on the system, fooling the system**
 - **New technology, is not available for every platform**
 - **Most commercial systems are targeted at Windows server market, leaving end users unprotected**



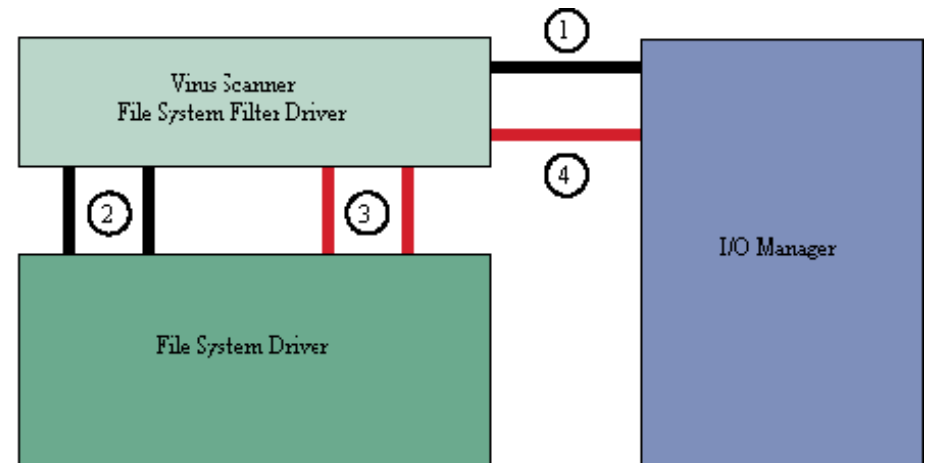
Current Anti-Virus Research

- **Behavior Blocking appears to be the future of Anti-Virus**
 - **Most research currently exploring various models of behavior blocking**
 - **Dr. Stephanie Forrest, University of New Mexico**
 - **IBM**
 - Both exploring possibility of using AI to model human immune system
 - **Entercept Corporation**
 - Developing solutions for Win2K servers
 - **Auburn University**
 - Beginning work on developing comprehensive behavior blocking for end users running Windows 2000/XP
 - **Initial Behavior Blocking software is out performing traditional virus scanners**
 - **Behavior Blocking software from Entercept successfully prevented infection by *Code Red & Nimda* during initial days of outbreak**



Research @ Auburn

- Auburn University “Filewall”
 - Abstract functionality very similar to a typical real-time virus scanner, or traditional firewall



- ① Original CreateFile IRP from I/O Manager
- ② CreateFile IRP propagated to file system
- ③ Virus scanner reads and checks file; if virus detected, virus scanner closes file
- ④ Virus scanner returns original CreateFile result, or failure if file contains virus



Research @ Auburn

- **Auburn University “Firewall”**
 - Geared towards end users running Windows 2000
 - Typical Operation:
 - **2 Examples:**
 - **Example 1:**
 - » User receives infected file (by e-mail, download, etc.)
 - » User executes infected file, which is infected by *VBS.Pie* (virus deletes all jpeg's from a user's computer)
 - » After executing, user is displayed with a warning, stating that the Windows Scripting Engine, while running *VBS.Pie* is attempting to delete files of type JPEG
 - » Hopefully, user is smart enough to realize that this is not proper, and kills the offending process



Research @ Auburn

- **Auburn University “Firewall”**
 - **Typical Operation: (con’t)**
 - **Example 2:**
 - » User is using Windows Explorer to view the contents of their disk.
 - » User discovers files that they no longer need, and are simply taking up space
 - » User decides to delete these files
 - » Firewall launches, and tells the user that *explorer.exe* is attempting to delete files of type <Whatever type of file is being deleted>
 - » User realizes this has been triggered by their own actions, and allows process to continue



Behavior Blocking

- **Large Anti-Virus companies have no current plans to offer Behavior Blocking software**
 - Network Associates (McAfee) & Symantec have no plans to offer any Behavior Blocking system in the coming year.
 - They claim this is because Behavior Blocking is not geared towards eradicating viruses once they are discovered

