

Principles of Information Security CS 5323 Lecture Eleven

Prof. William Winsborough
October 4, 2007

Business

- Recall: we will have an exam covering material we (read and) discussed in class October 11
 - This one will be worth about 20% of the final grade
- I will post a summary of the material that will be covered by sometime this weekend
- Tuesday October 9 will be devoted to discussion and review of exam material

4 October 2007

Winsborough CS 5323 Lecture 11

2

Unix Permissions

- Read, write, and execute for each of owner, group, world
 - For directories
 - Read: ability to list the directory
 - Write: adding and removing files in directory
 - Execute: make directory current working directory, open files inside the directory
- Additional bits
 - Set user ID on execution (bit in octal: 4000)
 - Set group ID on execution (bit in octal: 2000)
 - When sticky bit on a directory is set, files in the directory can be removed or renamed only by the owner of the file, who must also own the directory and have write permission on the directory (or by superuser) (bit in octal: 1000)

4 October 2007

Winsborough CS 5323 Lecture 11

3

Unix Command: chown

- Changes owner of a file
 - chgrp changes the group of a file
- Threat: attacker could create an SUID program and then change its owner to root
 - Some versions of Unix require root to chown
 - Others instead make chown turn off SUID and SGID bits
- Similar issues arise with chgrp

4 October 2007

Winsborough CS 5323 Lecture 11

4

Weakness of Unix Access Control

- Each object can have only one owner and one group
- Permissions control only read, write, and execute
- Other operations must be controlled using these basis permissions
 - System
 - Shut down system
 - Administer user accounts, groups
 - Application
 - High-level operations must be governed by using mechanisms implemented by the application

4 October 2007

Winsborough CS 5323 Lecture 11

5

General Security Principles

Higher-level Controlled Invocation

- Scenario:
 - Create a new UID “web server”
 - Owns resource and programs that access it
 - Permissions are granted to no other UID
 - Define all programs that access the resource to be SUID programs
- Threat:
 - Attacker may use any flaws in the SUID programs to gain control of resources owned by “web server”
 - This is an instance of the general problem that SUID programs (especially those owned by root) necessarily increase the size of the trusted computing base (TCB)

4 October 2007

Winsborough CS 5323 Lecture 11

6

Is File Deletion Secure?

- link and ln create multiple references to the same file
 - rm and rmdir remove a reference to a file
 - Files are only deallocated when *link counter* reaches zero
- Deallocation does not reinitialize disk block content
 - *wipe* does this
- Defragmentation can copy and deallocate blocks without wiping
 - Need a tool that initializes unallocated blocks

Access to Devices

- /dev
 - /dev/console console terminal
 - /dev/mem image of physical memory
 - /dev/kmem image of virtual memory
 - /dev/tty terminal
- Attackers can bypass controls on files if they can get access to the devices holding them
- Commands like ps (process status) have to have access to memory devices
 - Could make ps SUID to root, but creates vulnerability
 - Better: create a new group, mem, to own memory devices and make ps a SGID program

Mounting Threats

- The mount command links a new filesystem into Unix's main filesystem
- This can create vulnerabilities if, for instance, the new filesystem contains SUID to root programs of an attacker's own design
- To combat this threat, the mount command has flags and options such as:
 - Make files in the mounted filesystem non-executable
 - Turn off SUID and SGID bits on mounted filesystem

Trojan Horses

- Adding an attack program to a directory that occurs early in the search path and giving it the name of a popular command
- Defense: use full pathname to specify command, especially when running as root
- Don't put "." in your search path when you are in someone else's directory