CS590U
Access Control: Theory and Practice

Lecture 16 (March 9)

Confining UNIX Processes

Projects

- Project progress report due April 6
- Presentation during the last two weeks of class
  - April 17 to April 28
- No midterm
- Two more homeworks

What are some issues with UNIX access control?

- Coarse granularity (Windows the same)
  - access control is only per user
- Achieving least privilege is difficult

Best Practices for UNIX chroot() Operations

Steve Friedl's Unixwiz.net Tech Tips
http://www.unixwiz.net/techtips/chroot-practices.html

chroot

- The chroot system call **ch**anges the **root** directory of the current and all child processes to the given path, and this is nearly always some restricted subdirectory below the real root of the filesystem.
- The chroot system call almost all versions of UNIX, and it serves to create a temporary root directory for a running process, and it's a way of taking a limited hierarchy of a filesystem (say, /chroot/named) and making this the top of the directory tree as seen by the application.

Using chroot

- What are the security benefits?
  - under the new root, many system utilities do not exist, even if the attacker compromises the process, damage can be limited
- Examples of using chroot
  - ftp for anonymous user
- How to set up chroot?
  - need to set up the necessary library files, system utilities, etc., in the new environment

Can a process break out of chroot?

- Yes, usually need root privilege inside chrooted environment.

Jails: Confining the omnipotent root

Poul-Henning Kamp and Robert N. M. Watson

Jail is an extension of chroot implemented in FreeBSD

- chroot is limited
  - a process's visibility of the file system name-space is limited to a single subtree, but the process can still see/affect all other processes and networking spaces
- increasing the granularity of security controls increases the complexity of the administration process, in turn increasing both the opportunity for incorrect configuration, as well as demand on administrator time and resources
- need decentralized administration (partitioning), similar to a virtual machine

FreeBSD Jail syscall

- Restrictions in Jail
  - access to the file name-space is restricted in the style of chroot
  - the ability to bind network resources is limited to a specific IP address
  - the ability to manipulate system resources and perform privileged operations is sharply curtailed
  - the ability to interact with other processes is limited to only processes inside the same jail

Confining Root Programs with Domain and Type Enforcement

K.M. Walker et al. USENIX Seurity, 1996.

How does DTE work?

- objects are groups into types
- subjects are called domains
- define which types each domain has access to
- also define which domains can transit to which domains
  - only by executing one of entry programs into the domain
- init_domain determines the domain when system first starts
- assign statement binds types with files/directories

Protecting System Binaries from Rootkit

- Rootkits:
  - changing system binaries to add hidden backdoor and to hide the effects of break-in
- Strategy to defend against rootkit:
  - create a special administrative domain such that only this domain has write access to system binaries and the directories containing them
  - allow transition into these domain only

Thoughts

- A mechanism that is too flexible is not necessarily good

- Need to confine daemons
  - e.g., httpd
- Need to confine applications
- Need to confine unknown programs

Next Lecture

- Recent research in UNIX access control