1044414: Advanced Operating Systems and Virtualization **Spring 2018**

Lecture 19: May 15

Lecturer: Alessandro Pellegrini *Scribe: Anxhelo Xhebraj*

19.1 Baseline approaches to ensure Security

- Cryptography
- Authentication/Capabilities
- Security enhanced operating systems

We have already seen some approaches that try to mitigate problems.

Address Randomization: allows to run the same program in a different place of the virtual address space.

RIP-relative addressing mode allows to address variables by relative position from the `rip`.

19.2 User Authentication

The system must keep the password somewhere. Two files are used: `passwd` (legacy) and `shadow`.

The former was accessible by all the users while the latter only by root.

Traditionally there was the encrypted password using salt.

19.3 User IDs in Unix

Number used by the kernel to know which user is running a program. GID is used to identify the group to which the user running the program belongs to.

The administrator can rely on `su/sudo` that allows to run as `root` or any other user. `su [user]` where `user` is `root` if not specified.

Real user is the one that you login with. Effective tells the permissions. Saved tells which users you can become. SUID bits tell which users can run the program.

19.4 System calls for UID/GID system calls

This information is not only for "human" users. Root is UID 0. `geteuid` allow to get the *effective* user id. The kernel allows to run such syscalls only if you are already running as root.

`setuid` is not reversible. Will overwrite the user id not knowing which you were before. In this case the fact that usually each user has a shell associated to it when exiting such shell it gets back the user id (of the parent).

19.5 UNIX inetd

Controls services with specific port numbers.

References