Skip to content Skip to Navigation

My Professional Blog

Search

Search … Search

Tags

#32bit #64bit #Arch #architecture #ArchLinux #cpuaffinity #darkweb #deepweb #duckduckgo #facebooksecurity #findcmd #freeos #git #historyoflinux #ilovelinux #jenkins #linuxcommand #linuxcommandtips #linuxhistory #linuxtips #magicreboot #memorymanagement #memoryusage #onionsites #oracledb #os #securesocialmedia #semaphores #smp #sysrq #tor bare-metal k8s cluster bare-metal kubernetes cluster facebook k8s kubernetes kubernetes in raspberry pi 4 linux memory raspberrypi

Categories

DevOps FOSS Tools Linux Security Social

Content

Real, Effective & Saved UID explained

Each Linux/Unix process has 3 UIDs associated with it. Superuser privilege is UID=0.

Real UID

This is the UID of the user/process that created THIS process. It can be changed only if the running process has EUID=0.

Effective UID

This UID is used to evaluate privileges of the process to perform a particular action. EUID can be changed either to RUID, or SUID if EUID!=0. If EUID=0, it can be changed to anything.

Saved UID

If the binary image file, that was launched has a Set-UID bit on, SUID will be the UID of the owner of the file. Otherwise, SUID will be the RUID.

- **What is the idea behind this?**

Normal programs, like “ls”, “cat”, “echo” will be run by a normal user, under that users UID. Special programs that allow the user to have controlled access to protected data, can have Set-UID bit to allow the program to be run under privileged UID.

An example of such program is “passwd”. If you list it in full, you will see that it has a Set-UID bit and the owner is “root”. When a normal user, say “ajoy”, runs “passwd”, passwd starts with:

Real-UID = ajoy
Effective-UID = ajoy
Saved-UID = root

The program calls a system call “seteuid(0)” and since SUID=0, the call will succeed and the UIDs will be:

Real-UID = ajoy
Effective-UID = root
Saved-UID = root

After that, “passwd” process will be able to access /etc/passwd and change password for user “ajoy”. Note that user “ajoy” cannot write to /etc/passwd on it’s own. Note one other thing, setting a Set-UID on an executable file is not enough to make it run as a privileged process. The program itself must make a system call.

That is the idea.

#setuid#suidlinux

Published on:August 9, 2009 Category: Linux

«CPU affinity an overview »Kernel Modules

- ## Recent Posts
  - Understanding API
  - Site Reliability: SLI, SLO & SLA
  - Unikernel: Another paradigm for the cloud
  - Helm 3 – Sans tiller – Really?
  - Bare-Metal K8s Cluster with Raspberry Pi – Part 3
- About me
- Why this Blog?
- Visit My full profile

Powered by e.nigma and WordPress