

“Process user” control -- su, sudo, and SUID

David Morgan

© David Morgan 2003-08

Processes and users

- running processes are associated with user(s)
 - real user/UID –user ID of the process’ parent process
 - effective user/UID – determines resource access
 - real=“by whom” effective=“as whom” process runs
- process’s real and effective UIDs are same, usually
- login shell’s UIDs are (both) the one per the login
- check with `getuid()` and `geteuid()`

© David Morgan 2003-08

Process gets from caller/parent, gives to called/child

```
GETUID(2)                               Linux Programmer's Manual                               GETUID(2)
NAME
    getuid, geteuid - get user identity
SYNOPSIS
    #include <unistd.h>
    #include <sys/types.h>

    uid_t getuid(void);
    uid_t geteuid(void);
DESCRIPTION
    getuid() returns the real user ID of the current process.

    geteuid() returns the effective user ID of the current process.
```

© David Morgan 2003-08

Diagnostic UID revealer program

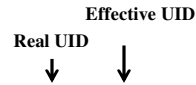
```
root@CHANG:~
[root@CHANG ~]# cat ids.c
/*
 * ids.c - Print UIDs and GIDs
 */
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>

int main(void)
{
    printf("Real user ID: %d\n", getuid());
    printf("Effective user ID: %d\n", geteuid());
    printf("Real group ID: %d\n", getgid());
    printf("Effective group ID: %d\n", getegid());
    exit(EXIT_SUCCESS);
}
```

another revealer: `ps -eo pid,ruid,euid,command`

© David Morgan 2003-08

ps -eo pid,ruid,euid,command



```
root@c1ay:~/class/books/wall/Chap04
[root@c1ay Chap04]# ps -eo pid,ruid,euid,command | tail -17
11837 0 0 sshd: root@pts/0
11839 0 0 -bash
11936 500 500 -bash
11964 500 500 xinit -- :1
11965 500 0 X :1
11986 500 500 xterm -geometry +1+1 -n login
11988 500 500 bash
12003 500 500 /bin/sh /usr/lib64/firefox-1.5.0.7/firefox -UI
12014 500 500 /bin/sh /usr/lib64/firefox-1.5.0.7/run-mozilla
ox-bin -UILocale en-US
12019 500 500 /usr/lib64/firefox-1.5.0.7/firefox-bin -UILoca
12024 500 500 /usr/libexec/gconfd-2 11
12237 503 503 -bash
12263 503 0 passwd
12283 0 0 [pdflush]
12287 0 0 [pdflush]
12292 0 0 ps -eo pid,ruid,euid,command
12293 0 0 tail -17
[root@c1ay Chap04]#
```

3 active users, 0 500 503
2 processes have effective UID different from real UID

© David Morgan 2003-08

Controlling a process's UIDs

- su
- sudo
- SUID

© David Morgan 2003-08

su syntax

su <-c command> <user>

- defaults
 - omit user: root
 - omit command: bash
- password prompt: for *other user's* password, not yours

© David Morgan 2003-08

Run “ids”

Q: where does it pick up its UIDs from?

A: inherits them

- run it from 2 diff logins
 - reflects the logins' UIDs
- su to some other user, then run it
 - reflects the other user's UIDs
- run fork9a mini-shell via su as another user, then run ids from fork9a
 - fork9a reflects the UIDs of the launching shell
 - ids in turn reflects those of fork9a

© David Morgan 2003-08

sudo – secure solution

- lets certain user(s) run certain program(s) as another user
- user runs program indirectly under sudo's control: `sudo <targetprogram>`
- sudo configuration defines who can run what as whom

© David Morgan 2003-08

sudo syntax

`sudo <-u user> command`

- defaults
 - omit user: root
 - omit command: *not optional*
- password prompt: for *your* password, not other user's (you don't know who that is)

© David Morgan 2003-08

sudo config file: /etc/sudoers

- privilege specifications
- other specifications
 - User aliases – named groups of “by” users
 - Runas aliases – named groups of “as” users
 - Cmnd aliases – named groups of commands

© David Morgan 2003-08

What?

- What is a “ ‘by’ user” ?? an “ ‘as’ user” ??
- default: command runs as whoever launched it
- sudo purpose: let a command launched by one user run as another
 - “by” user is the one who launches the command
 - “as” user is the one the command runs as, as if he had actually launched it (even though he didn't)

© David Morgan 2003-08

sudoers privilege specifications

<who by> <where>=(<who as>) <what>

© David Morgan 2003-08

SUID – exception to the rule

- SUID – a permission characteristic of files
- changes the effective UID of file's process when run
 - from UID of user who runs the program
 - to UID of user who “owns” the file

© David Morgan 2003-08

SUID – exception to the rule

- WHAT – “When a SUID file is run, the process involved takes on an *effective UID* that is the same as the owner of the file.”
- WHY – “Sometimes, unprivileged users must be able to accomplish tasks that require privileges.
- EXAMPLES – passwd, mail

© David Morgan 2003-08

passwd uses SUID

passwd program's executable is SUID
therefore runs as whichever
user is its owner

```
root@clay:~  
[root@clay ~]# /bin/ls -l /usr/bin/passwd  
-rwsr-xr-x 1 root root 27768 Jul 17 2006 /usr/bin/passwd  
[root@clay ~]#  
[root@clay ~]# /bin/ls -l /etc/shadow  
----- 1 root root 1393 Jan 10 11:29 /etc/shadow  
[root@clay ~]#  
[root@clay ~]#
```

which in the case of password database /etc/shadow,
since it can be read by its owner who is root,
enables the passwd program to read /etc/shadow...

... which lets users change their own passwords

© David Morgan 2003-08

Applying SUID to a file

```
root@clay:~# ls -l testfile
-rwxr-xr-x 1 root root 4 Jan 12 11:09 testfile
root@clay ~]#
root@clay ~]# chmod u+s testfile
root@clay ~]#
root@clay ~]# ls -l testfile
-rwsr-xr-x 1 root root 4 Jan 12 11:09 testfile
root@clay ~]#
```

© David Morgan 2003-08

Effect of SUID

ids program reports
as whom it is running

```
root@clay:~# whoami; ls -l ids
root
-rwxr-xr-x 1 root root 6337 Jan 12 12:11 ids
root@clay ~]#
root@clay ~]# ./ids
Real user ID: 0
Effective user ID: 0
Real group ID: 0
Effective group ID: 0
root@clay ~]#
root@clay ~]# chown david ids
root@clay ~]# chmod u+s ids
root@clay ~]# ls -l ids
-rwsr-xr-x 1 david root 6337 Jan 12 12:11 ids
root@clay ~]#
root@clay ~]# ./ids
Real user ID: 0
Effective user ID: 500
Real group ID: 0
Effective group ID: 0
root@clay ~]#
```

run in this root shell,
ids runs as root (gets it
from the shell)

but if its executable is SUID
and david owns it,
ids runs as david (gets it
from the executable)

© David Morgan 2003-08

SUID shell scripts

- BAD
 - DON'T
 - Security flaw – launches SUID shell to run script
-
- most modern unix's now ignore SUID on a script

© David Morgan 2003-08