Permisos

Seminario de Desarrollo de Software - Casa Central.

Maximiliano Osorio
mosorio@inf.utfsm.cl

Universidad Técnica Federico Santa María

24 de septiembre de 2017

Usuarios

- Cada programa corre como un usuario particular.
- Cada archivo es propiedad de un usuario.
- Acceso a archivos y directorios están definidos por un usuario.
- Los usuarios son cuentas asociadas a humanos o aplicaciones.
- Identificados por UID

```
[mosorio@ssh ~]$ id
uid=5844(mosorio) gid=5844(mosorio) groups
   =5844(mosorio) context=unconfined_u:
   unconfined_r:unconfined_t:s0-s0:c0.c1023
```

Permisos y archivos

```
[mosorio@ssh ~]$ ls -l
drwxr-xr-x   12 mosorio  staff         408 Apr  6
    14:29 repos
-rw-r--r--    1 mosorio  staff          22 Mar 14
    23:05 vtr
```

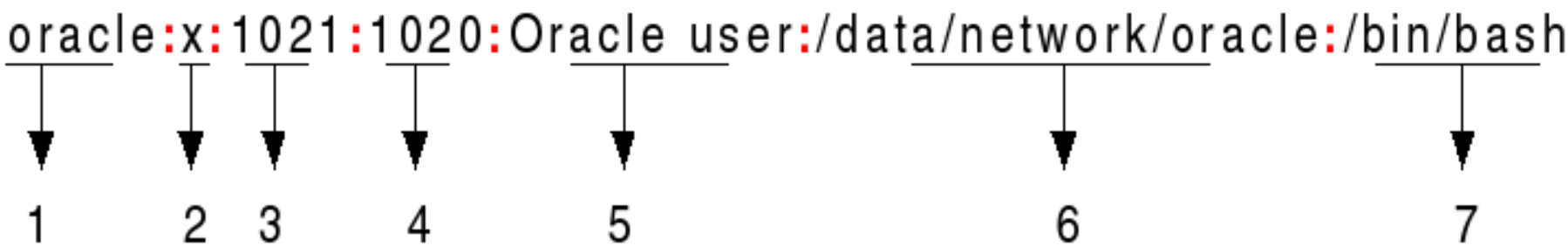
Permisos y procesos

```
[mosorio@ssh ~]$ ps au
USER        PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root       1726  0.0  0.0   4068   584 tty1     Ss+  Jan26   0:00 /sbin/mingetty
root       1729  0.0  0.0   4068   584 tty2     Ss+  Jan26   0:00 /sbin/mingetty
root       1731  0.0  0.0   4068   580 tty3     Ss+  Jan26   0:00 /sbin/mingetty
root       1733  0.0  0.0   4068   580 tty4     Ss+  Jan26   0:00 /sbin/mingetty
root       1735  0.0  0.0   4068   580 tty5     Ss+  Jan26   0:00 /sbin/mingetty
root       1752  0.0  0.0   4068   584 tty6     Ss+  Jan26   0:00 /sbin/mingetty
mosorio   13000  1.4  0.3 108172  3692 pts/1    Ss   23:50   0:00 -bash
mosorio   13128  0.0  0.1 108352  1132 pts/1    R+   23:50   0:00 ps au
ydossow   13280  0.0  0.2 106696  2080 pts/0    Ss+  Mar08   0:01 -bash
```

/etc/passwd

Existe un archivo que guarda la información de los usuarios:**/etc/passwd**

1. Username: Nombre
2. Password placeholder
3. User ID (UID): Identificador único del usuario
4. Group ID (GID): Identificador único del grupo
5. User ID Info: Información extra del usuario, ej. nombre completo.
6. Home directory: Ruta absoluta del home del usuario.
7. Command/shell: Ruta absoluta del comando o shell (/bin/bash).



UID

Los UID tienen un estandar:

- UID 0 (zero) es root.
- UID 1-99 están reservados para cuentas predefinas.
- UID 100-999 están reservadas para cuentas de sistemas.

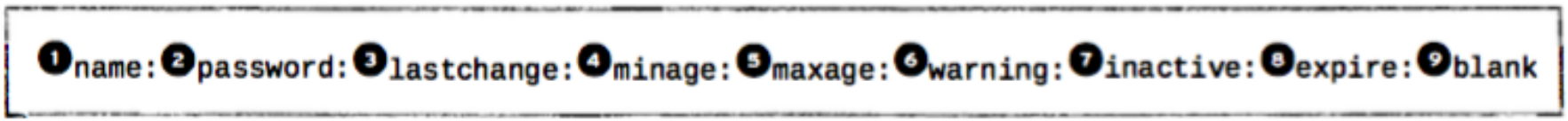
Es común ver usuarios con shell **/sbin/nologin** para prevenir que no existan logeo.

Grupos

Como los usuarios, cada grupo tiene un nombre y id (GID), se definen en **/etc/group**

- Cada usuario tiene un **grupo primario**.
- La relación se puede encontrar en la tercera columna de /etc/passwd.
- Existen grupos extras que se encuentran en /etc/group.
- Un usuario siempre tiene un grupo, este grupo es llamado primario y tiene el mismo nombre.

/etc/shadow



- 1. login name.
- 2. Encrypted password.
- 3. Fecha del ultimo cambio de password.
- 4. Mínimos de número de días para que password pueda cambiar.
- 5. Máximo de número de días para la password deba ser cambia.
- 8. Fecha de expiración

root user

Todos los sistemas tiene un usuario del tipo superuser

- superuser: es un usuario que tiene poder supremo y total del sistema.
- Con este gran privilegio viene una gran responsabilidad.

su -

- El comando su permite cambiar a otra cuenta, si el usuario no se encuentra especificado en el comando, por defecto es root.
- El comando **su username** inicia un shell con las configuraciones de ambientes del usuario que lanza el comando.
- El comando **su - username** inicia un shell con las configuraciones de ambientes del usuario **username**

Warning

- Con *su -* se puede suplantar a un usuario.
- Usar **/bin/su** es recomendado.

sudo

- **sudo** toma un argumento y lo ejecuta como root
- **sudo** con consulta a **/etc/sudoers** para conocer quien está autorizado.
- Cada cinco minutos preguntará la clave.

sudo

- La desventaja de usar **su** es que la cuenta tiene todos los privilegios.
- **sudo** permite acceder tareas de administración en base al archivo **/etc/sudoers** y se debe ingresar la *password*.
- Permite manejar con granularidad en los permisos.
- Al utilizar sudo queda un log del comando

```
[root@yanara ~]# tail -f /var/log/secure
Oct 14 16:09:45 yanara sudo: linets : TTY=pts/1
    ; PWD=/home/linets ; USER=root ; COMMAND=/
   sbin/ip a
Oct 14 16:09:47 yanara sudo: linets : TTY=pts/1
    ; PWD=/home/linets ; USER=root ; COMMAND=/
   bin/su -
Oct 14 16:09:47 yanara su: pam_unix(su-l:
   session): session opened for user root by
   linets(uid=0)
```

sudo

El usuario root puede ejecutar desde todas las terminales actuando como ALL (any) usuarios y correr ALL (any) comando.

```
root  ALL=(ALL)  ALL
```

sudo

El usuario operator puede correr **power off** de cualquier terminal.

```
operator ALL= /sbin/poweroff
```

Manejando cuentas locales

Utilities	Description
id	Displays user and group IDs.
useradd, usermod, userdel	Standard utilities for adding, modifying, and deleting user accounts.
groupadd, groupmod, groupdel	Standard utilities for adding, modifying, and deleting groups.
gpasswd	Standard utility for administering the /etc/group configuration file.

Para conocer más de los comandos asociados, se recomienda consultar RHEL Doc: Crear, eliminar y manejo de grupos

useradd

Crear una usuario y setea los campos en **/etc/password**. No setea una password

```
useradd mosorio
```

userdel

- **userdel username** remueve al usuario pero deja el home intacto.
- **userdel -r username** y el home.

userdel

Cuando un usuario es eliminado sin la opción -r, el sistema deja los archivos con UID no asignado.

leak

```
[root@serverX ~]# useradd prince
[root@serverX ~]# ls -l /home
drwx------. 3 prince  prince    74 Feb  4 15:22 prince
[root@serverX ~]# userdel prince
[root@serverX ~]# ls -l /home
drwx------. 3    1000    1000   74 Feb  4 15:22 prince
[root@serverX ~]# useradd bob
[root@serverX ~]# ls -l /home
drwx------. 3 bob     bob       74 Feb  4 15:23 bob
drwx------. 3 bob     bob       74 Feb  4 15:22 prince
```

```
find / -nouser -o nogroup 2> /dev/null
```

Quitando acceso

```
[mosorio@ip122 ~]$ sudo usermod -L vonbrand
[mosorio@ip122 ~]$ su - vonbrand
Password:
su: Authentication failure
```

Linux file system permissions

El acceso a los archivos son controlados por Linux.

- Grupos de permisos:
  - El archivo tiene un dueño.
  - El archivo tiene un grupo.
  - Se puede definir permisos para el dueño, grupo y todos los demás usuarios.
- Tipos permisos:
  - read: Permite leer el archivo
  - write: Permite escribir o modificar el archivo.
  - execute: Permite ejecutar el archivo o ver los contenidos de un directorio.

Precedencia

Existe precedencia, primero se ve owner, después group y luego others.

Viendo los permisos

Permisos

r: read. **w:** write. **x:** execute.

```
drwxrwxr-x.   2 mosorio mosorio   taller_redes
drwx------.   2 mosorio mosorio   Templates
drwxr-xr-x.   6 mosorio mosorio   term.js
-rw-rw-r--.   1 mosorio mosorio   testing
-rw-rw-r--.   1 mosorio mosorio   test.sh
drwxrwxr-x.   2 mosorio mosorio   tmp
-rw-rw-r--.   1 mosorio mosorio   Tor_ip_list
```

Método simbolico

```
chmod WhoWhatWhich file|directory
```

- who es u,g,o,a (user, group, other, all)
- what es +,-,= (add, remove, exactly)
- which es r,w,x (read, write, executable)

Método numerico

Cada digito representa un nivel de acceso: user, group, other

```
chmod ### file|directory
```

- # es la suma de r=4, w=2 y x=1.

Ejemplos

- Remueve los permisos de lectura y escritura para el grupo y otros sobre el file1

```
chmod go-rw file1
```

- Añade permisos de ejecución al archivo 2.

```
chmod a+x file2
```

- Setea permisos todos los permisos para el dueño, lectura y ejecución para el grupo y nada para otros.

```
chmod 750 sampledir
```

Recursividad

chmod -R, permite cambiar los permisos recursivamente.

Cambiando dueños

Por defecto cuando se crea un archivo o directorio el dueño del éste, es quien ejecutó el comando.
Para cambiar el dueño se utiliza **chown**.

```
chown student folder
```

```
chown student:student folder
```

```
chown :student folder
```

```
chown -R student:student folder
```

setuid

- setuid: significa que el **command** va correr como el dueño del archivo.

```
[mosorio@ssh ~]$ ls -l /usr/bin/passwd
-rwsr-xr-x. 1 root root 30768 Feb 17  2012 /
    usr/bin/passwd
```

sticky

- sticky: se setea a un directorio y significa que solo el usuario dueño y root pueden borrar el archivo.

```
[mosorio@ssh ~]$ ls -ld /tmp/
drwxrwxrwt. 3 root root 4096 Oct 14 18:55 /
    tmp/
```

setgid

- En archivos: se ejecutan como el dueño.
- En directorios: los archivos nuevos en el directorio tienen de dueño al dueño del directorio.

setuid, setgid, sticky

- Simbolicamente: u+s, g+s, o+t
- x: 4###, 2###, 1###

```
chmod g+s directory
chmod 2700 directory
```

Procesos

- Todo proceso y archivos tiene un usuario relacionado.
- El usuario está asociado al proceso y determina los archivos y directorios que son accesibles.

```
[mosorio@ssh ~]$ ps au
USER        PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root       2091  0.0  0.0   4064   348 tty1     Ss+  Aug26   0:00 /sbin/mingetty /dev/
     tty1
root       2093  0.0  0.0   4064   348 tty2     Ss+  Aug26   0:00 /sbin/mingetty /dev/
     tty2
root       2095  0.0  0.0   4064   352 tty3     Ss+  Aug26   0:00 /sbin/mingetty /dev/
     tty3
root       2097  0.0  0.0   4064   348 tty4     Ss+  Aug26   0:00 /sbin/mingetty /dev/
     tty4
root       2099  0.0  0.0   4064   348 tty5     Ss+  Aug26   0:00 /sbin/mingetty /dev/
     tty5
root       2101  0.0  0.0   4064   348 tty6     Ss+  Aug26   0:00 /sbin/mingetty /dev/
     tty6
mosorio   26034  0.1  0.3 108156  3696 pts/2    Ss   20:22   0:00 -bash
mosorio   26238  0.0  0.1 108340  1108 pts/2    R+   20:24   0:00 ps au
ydossow   26794  0.0  0.1 106544  1920 pts/0    Ss   Oct09   0:00 -bash
ydossow   27246  0.0  0.3  60472  3864 pts/0    S+   Oct09   0:08 ssh root@fw
ydossow   28304  0.0  0.1 106544  1976 pts/1    Ss+  Oct09   0:00 -bash
```