SEGURIDAD A NIVEL PERMISOS DE ARCHIVOS BASICOS, ESPECIALES Y ACL

por Ronaldo Cabezas

Configuración básica de archivos

Octal	Decimal	Permission	Representation
000	0 (0+0+0)	No Permission	---
001	1 (0+0+1)	Execute	--x
010	2 (0+2+0)	Write	-w-
011	3 (0+2+1)	Write + Execute	-wx
100	4 (4+0+0)	Read	r--
101	5 (4+0+1)	Read + Execute	r-x
110	6 (4+2+0)	Read + Write	rw-
111	7 (4+2+1)	Read + Write + Execute	rwx

Numeric	Permission Type	Permission To
400	Read	Owner
040	Read	Group
004	Read	Others
200	Write	Owner
020	Write	Group
002	Write	Others
100	Execute	Owner
010	Execute	Group
001	Execute	Others

chmod 600 file
chown user file
chgrp group file

SEGURIDAD A NIVEL PERMISOS DE ARCHIVOS BASICOS, ESPECIALES Y ACL

por Ronaldo Cabezas

 # chown radiusd.radiusd -R *

por Ronaldo Cabezas

Atributos Especiales (ext3, etx4)

$> su -

$> vim prueba
hola mundo

$> chattr +i prueba
añade el bit de inmutabilidad, el archivo no puede ser modificado

$> lsattr prueba
lista los atributos de un archivo

$> vim prueba
hola mundo que tal
ESC:wq!

SEGURIDAD A NIVEL PERMISOS DE ARCHIVOS BASICOS, ESPECIALES Y ACL

por Ronaldo Cabezas

$> chattr -i prueba
quita el bit de inmutabilidad

$> vim prueba
hola mundo que tal
ESC:wq

$> chattr +a prueba
establece que el archivo solo se puede escribir añadiendo contenido

$> chattr -a prueba
quita la opción de solo añadir

$> man chattr
para ver lista de atributos

por Ronaldo Cabezas

Permisos Especiales:

SUID
set UID
El bit SUID activo en un archivo significa que el que lo ejecute va a tener los mismos permisos que el que creó el archivo.

SGID
set GID
El SGID se da a nivel de grupo. Es decir, todo archivo que tenga activo el SGID, al ser ejecutado, tendrá los privilegios del grupo al que pertenece.

Sticky
bit de persistencia
El Sticky bit se utiliza para permitir que cualquiera pueda escribir y modificar sobre un archivo o directorio, pero sólo su propietario o root pueda eliminarlo.

por Ronaldo Cabezas

Nivel especial

Sólo root puede asignar este bit

0 ---> ningún permiso, valor por defecto

1 ---> sticky

2 ---> sgid

3 ---> sgid+sticky

4 ---> suid

5 ---> suid+sticky

6 ---> suid+sgid

7 ---> suid+sgid+sticky

Setuid/Setgid/Sticky bit

por Ronaldo Cabezas

read/setuid	write/setgid	execute/sticky
4	***2***	***1***

Special	User	Group	Other
7	***7***	***7***	***7***
	rwx	rwx	rwx
	rws	**rws**	**rwt**

chmod 7777 archivo

chmod u=rwxs, g=rwxs, o=rwxt archivo

Setuid/Setgid/Sticky bit

por Ronaldo Cabezas

read/setuid	write/setgid	execute/sticky
4	***2***	***1***

Special	User	Group	Other
7	***6***	***6***	***6***
	rw-	rw-	rw-
	rwS	**rwS**	**rwT**

chmod 7666 archivo

chmod u=rws, g=rws, o=rwt archivo

Setuid/Setgid/Sticky bit

por Ronaldo Cabezas

read/setuid	write/setgid	execute/sticky
4	***2***	***1***

Special	User	Group	Other
5	***7***	***3***	***1***
	rwx	-wx	- -x
	rws	**-wx**	**- -t**

chmod 5731 archivo

chmod u=rwxs, g=wx, o=xt archivo

Setuid/Setgid/Sticky bit

por Ronaldo Cabezas

read/setuid	write/setgid	execute/sticky
4	***2***	***1***

Special	User	Group	Other
7	***6***	***7***	***0***
	rw-	rwx	- - -
	rwS	**rws**	**- -T**

chmod 7670 archivo

chmod u=rws, g=rwxs, o=t archivo

SEGURIDAD A NIVEL PERMISOS DE ARCHIVOS BASICOS, ESPECIALES Y ACL

por Ronaldo Cabezas

chmod

chmod 5731 archivo

ls -l archivo

rws - wx --t archivo

chmod u=rws,g=rwxs,o=xt archivo

ls -l archivo

rwS rws --t archivo

chmod a+xs archivo

ls -l archivo

rws -ws --t archivo

Ejemplo de Sticky

(Bloqueando borrado de archivos)

por Ronaldo Cabezas

$> su –

$> mkdir /a

$> chmod 1777 /a

Dando permisos de sticky al directorio;
sólo root podrá borrar los archivos del directorio; aunque estos tengan permisos totales

$> echo "date" > /a/lahora

$> chmod 777 /a/lahora

le damos permisos totales al archivo lahora

$> ls -ld /a

se verifica el sticky del directorio

$> exit

Como alumno intentar borrar lahora

$> rm /a/lahora

no puede borrar el archivo ya que el directorio con sticky protege a los archivos, sólo root puede borrar

Ejemplo de SUID
(ejecutando un archivo como el dueño)

por Ronaldo Cabezas

$> su –

$> chmod 4755 /usr/bin/vim
Asignando el SUID al editor de texto vim
Loguearnos como usuario

$> su - alumno
Editar archivo /etc/hosts, agregando al final, grabar con
ESC:wq!

$> vim /etc/hosts
#Colocando ip y dominio de mensajería
El usuario alumno pudo grabar en el archivo /etc/hosts porque ejecuto vim como si fuera root

$> su –

$> chmod 0755 /usr/bin/vim
Reestablece los permisos originales y le quita el SUID

Ejemplo de SGID (directorio de grupo)

por Ronaldo Cabezas

```
$> groupadd sistemas
$> mkdir /sistemas
$> chgrp sistemas /sistemas
$> chmod 2770 /sistemas
$> gpasswd -a alumno sistemas
$> gpasswd -a tuxito sistemas
$> su - alumno
$> touch /sistemas/unarchivo
$> exit
$> su - tuxito
$> touch /sistemas/otroarchivo
$> exit
$> su - root
$> touch /sistemas/nuevoarchivo
$> ls -l /sistemas ---> se observa que los archivos se crearon con grupo sistemas
```

SEGURIDAD A NIVEL PERMISOS DE ARCHIVOS BASICOS, ESPECIALES Y ACL

por Ronaldo Cabezas

ACL

```
$> su -
```

```
$> useradd -md /home/tuxito tuxito
```
crea un nuevo usuario, su directorio de trabajo y su grupo igual

```
$> passwd tuxito
```
crea o cambia la contraseña de un usuario

```
$> su - tuxito
```
se loguea como tuxito

```
$> id
```
muestra información del usuario actual

```
$> exit
```
sale de la sesión

```
$> id
```

SEGURIDAD A NIVEL PERMISOS DE ARCHIVOS BASICOS, ESPECIALES Y ACL

por Ronaldo Cabezas

$> usermod -g users -G video,audio -s /bin/bash tuxito

agregamos al grupo primario users a tuxito y al secundario video y audio

$> setfacl -m u:alumno:rX /home/tuxito

configuramos los permisos del usuario tuxito lectura y ejecución

$> getfacl /home/tuxito

verificamos los permisos del usuario tuxito