GUÍA 9: CONTROL DE ACCESOS – SOLUCIONES

Ejercicio 1:

La matriz de control de accesos es:

	X	Y	Z
Alice	read write	read	execute
Bob	read	read write	-

a) Listas de control de acceso:

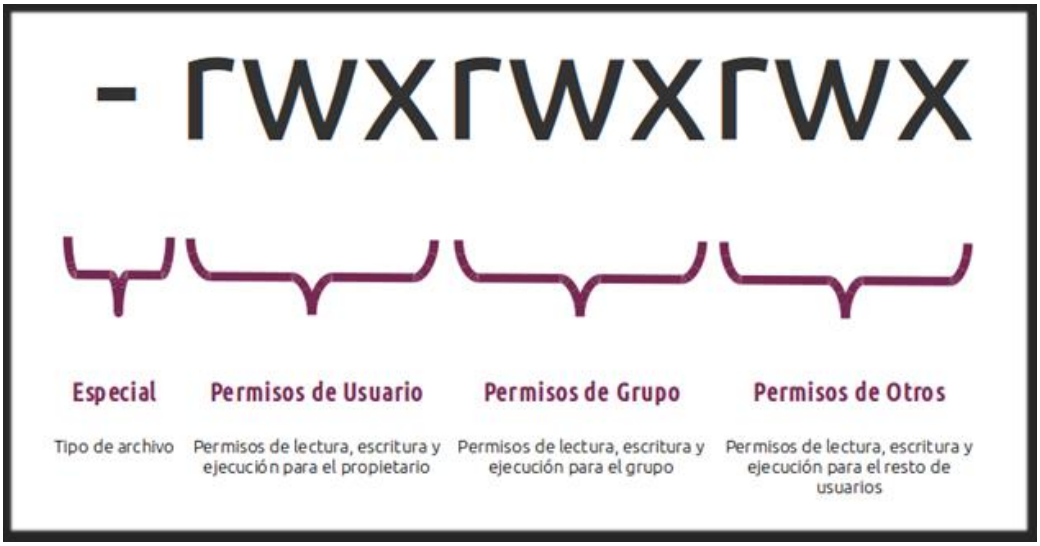
ACL(x) = {(Alice, r w), (Bob, r)}
ACL(y) = {(Alice, r), (Bob, rw)}
ACL(z) = {(Alice, x)}

b) Listas de capacidades

C-list(Alice) = {(x, rw), (y, r), (z, x)}
C-list(Bob) = {(x, r), (y, rw)}

c) Si quiere revocar los accesos de Alice a cualquier archivo, entonces se borra la lista de capacidades de Alice. Si se quiere borrar accesos de Alice para algunos archivos específicos, se borran las entradas de Alice dentro de la lista de control de acceso de esos objetos.

Ejercicio2:



	File1	File2	File3
dick	rwxo	-	rx
Jane	rx	rw	orwx
sally	r	-	rx
Root	rwx	rwxo	rwx

a) El usuario root sólo puede hacerlo. Sólo el usuario root puede cambiar el grupo de un archivo o directorio.
b) Quita todos los permisos a todos los otros (usuarios del grupo y owner queda igual)

El comando chmod tiene los siguientes parámetros:

```
$ chmod [u|g|o|a][+|-][r|w|x] fichero
```

donde:
u = user
a = all
g = group
o = others

+ = habilita permiso
- = quita permiso

c) Da permiso de lectura a todos.

Ejercicio3:

a) Matriz de acceso:

	.	..	Docs	Group	howto	Private	Private.tar
Hm	**rwxo**	**rx**	**rwx**	**rwx**	**rw**	**rwx**	**rw**
Alice	**rwx**	**rx**	**rx**	**rwx**	*****	**--**	**rw**
Bob	**rwx**	**rx**	**rx**	**rx**	**r**	**--**	**rw**
root	**rwx**	**rwxo**	**rwx**	**rwx**	**rwx**	**rwx**	**rwx**

*

Como Alice pertenece a infsec, no tiene permiso de nada sobre HOWTO. (Porque en los permisos de grupo que vemos en el listado que sale con ls dice: ---) Si Alice no perteneciera al grupo infsec, le correspondería el permiso de "others", y ahí sí tendría "r", que es lo que ocurre con Bob. Como Alice está en infsec, entonces ya no se analiza "others" al tener inhabilitados los permisos de grupo.

b) Listas de control de acceso:

Mirando la matriz, se toman las columnas, armando pares (sujeto, permisos)
ACL(Docs) = {(hm,rwx), (Alice, rx),(Bob,rx),(root,rwx)}
O también teniendo en cuenta los grupos:
ACL(Docs)={(hm,rwx), (Infsec,rx),(Otros,rx),(root,rwx)}

c) Alice quiere ver archivos de private.tar:

Para ver los archivos de private.tar, debería poder descomprimir en carpeta donde tenga permiso para ejecutar y escribir.
Como todos pueden escribir en directorio actual, Alice puede descomprimir private.tar en directorio actual (puede hacerlo en un directorio que cree para eso, para que no coincida con private) y ver lo que tiene (sin entrar al directorio private)
También podría copiar private.tar en Group y hacerlo dentro de Group.

Ejercicio 4:

a)

	O1	O2	O3
Alice	rw	-	-
Bob	rw	-	rw
Carol	-	rw	-
Dave	-	-	rw
Ellen	r	r	-

b)
ACL(o1 = diseños) = {(ing, rw), (CEO, r)}
ACL(o2 = documentos financieros) = {(finanzas, rw), (CEO, r)}
ACL(o3 = archivos configuración) = {(admin., rw)}

c) C-List(Alice) = {(o1, rw)}
C-List(Bob) = {(o1, rw), (o3, rw))}
C-List(Carol) = {(o2, rw)}
C-List(Dave) = {(o3, rw)}
C-List(Ellen) = {(o1, r), (o2, r)}

d) Habría que agregar derechos de “take ownership” y “delegate/grant” a Ellen. Si el sistema operativo no lo permite, otra forma es que Ellen tenga dos tipos de login.

Ejercicio 5:

a) Pueden ser ambos, aunque quizás sea más cómodo C-Lists.

a) Se pueden hacer dos C-Lists. Con ACL podría ser más fácil ver, para un determinado archivo si un usuario puede acceder o no .

b) Conviene hacer ACL para cada xi que se desea hacer público: ACL(xi) = {(*,r)}

Ejercicio 6:

El extranjero es el (5,1)

Ejercicio 7:

Distribuir las partes de un esquema (30,10) dándole al general 10 partes, a cada coronel 5, a los otros 2 a cada uno. Así se necesitan 10 para descubrir el secreto.