De Manuais Referencia Departamento Informatica

Nesta sección superaranse as limitacións do tradicional sistema de permisos de UNIX. Veranse as Listas de Control de Acceso (ACLs), que permiten indicar que conxunto de grupos e/ou usuarios poden acceder a un arquivo e/ou carpeta , sen estar limitados a Usuario Propietario, Grupo Propietario e Outros.

As ACLs aplicadas sobre un sistema arquivos exportado por NFS, tamén son efectivas nos clientes.

LEMBRAR EN UBUNTU DESKTOP

En Ubuntu Desktop, hai unha introdución ó sistema de arquivos en canto ós permisos e a propiedade das carpetas e arquivos.

Recoméndase que se revisen as seccións

Usuarios e grupos en Ubuntu

- **Sistema de arquivos e carpetas**
  - **Nautilus e carpeta persoal de usuario**
  - **Operacións con discos e soportes externos: montar, desmontar, formatar, etc.**
  - **A xeraquía dos sistema de ficheiros de GNU/Linux**
- **Permisos básicos de ficheiros e carpetas**
- **ACLs**

do curso Curso Platega 08-09: Sistema operativo GNU-LINUX: UBUNTU 8.10.

Contenido

Introdución ás ACLs

O sistema tradicional de permisos de GNU/Linux só permite afinar os permisos dunha carpeta ou arquivo para o usuario e o grupo propietario dos mesmos. Para os demais queda un grupo chamado *outros* que non permite facer ningún axuste fino.

Permisos básicos

Existen tres permisos independentes, chamados **permisos básicos**, que poden ser permitidos (estado 1) ou denegados (estado 0) a un arquivo e/ou directorio.

- **r - lectura**
- **w - escritura**
- **x - execución**

O significado destes tres permisos resúmese na seguinte táboa:

Permiso	Arquivo	Directorio
Lectura	Ver o contido do arquivo.	Ver o nome dos arquivos dentro do directorio (pero sen poder saber nada máis sobre eles como: o tipo de arquivo, tamaño, propietario, permisos, etc.)
Escritura	Modificar ou eliminar el arquivo.	Agregar, eliminar e renomear arquivos do directorio
Execución	Executar o arquivo.	Percorrer o súa árbore para acceder arquivos e subdirectorios, pero non velos arquivos dentro do directorio (excepto que se lle dea o permiso de lectura)

Clases de usuarios

Clases de usuarios

Os permisos de sistemas UNIX divídense en catro *clases*, coñecidas como *usuario*, *grupo*, *outros* e *todos* (con frecuencia abreviado *UGOA* polas súas siglas en inglés).

Por lo tanto, as clases de usuarios ás cales se lles poden asignar os **permisos básicos** anteriormente mencionados son:

- **u – dono**: dono do arquivo ou directorio
- **g – grupo**: grupo ó que pertence ó arquivo
- **o – outros**: todos os demais usuarios que no son o dono nin do grupo
- **a – todos**: inclúe ó dono, ó grupo e a outros

Os *permisos efectivos* aplicados a un determinado usuario en relación a un arquivo determínanse nunha orde lóxica de precedencia. Por exemplo,ol usuario propietario del arquivo terá os permisos efectivos dados á clase de usuario, sen importar os asignados a clase do grupo ou a la clase de outros.

ACLs

O sistema de permisos básicos so permite afinar permisos sobre obxectos para o usuario propietario e grupo propietario, os demais usuarios e grupos van nun *feixe* todos xunto na clase *outros*.

As Listas de Control de Acceso (ACLs)
permiten un acceso máis granular ó sistema de arquivos de GNU/Linux, permitindo indicar que varios usuarios e varios grupos teñen acceso en modo lectura a unha carpeta, por exemplo. Tamén se pode indicar cales deses usuarios ou grupos se deben herdar para os obxectos secundarios (ficheiros e subcarpetas) que se creen dentro desa carpeta.

As acls, por agora non veñen de **serie** coa instalación de GNU/Linux. Haberá que instalar o paquete e configurar o sistema de arquivos sobre o que se desexan aplicar.

Imaxinar unha carpeta chamada **asi** (Ciclo de administración de sistemas informáticos) que alberga as carpetas home de cada alumno dese ciclo. Interesa que a esa carpeta só entre o root (control total) e os alumnos de asi, para logo que cada quen acceda á súa carpeta particular. Finalmente os profesores dese curso poden entrar na carpeta asi, e en tódalas subcarpetas e ficheiros (actuais e futuros) en modo lectura.

Para iso esa carpeta **asi**, terá a seguinte acl:

```
# file: asi                    -- carpeta/ficheiro sobre o que hai unha acl.
# owner: root                  -- usuario propietario: root
# group: root                  -- grupo propietario: root
```

```
# group: root                    -- grupo propietario: root
user::rwx                        -- Permisos do usuario propietario: rwx
group::---                       -- Permisos do grupo propietario: ningún
group:g-asi-alum:r--             -- Grupos que hai na acl: g-asi-alum:r . Este grupo non se herdará a
group:g-asi-profes:r--           -- Grupos que hai na acl: g-asi-profes:r . Este grupo herdarase a ob

mask::r--
other::---                       -- Permisos da clase outros: ningún
                                 -- Nas seguintes substitúase default por herdar e entenderase mellor
                                    E pénsese nun obxecto secundario (subcarpeta ou ficheiro)
                                    que se cre dentro da da carpeta asi.

default:user::rwx                      -- Permisos que herdará o usuario propietario: rwx
default:group::---               -- Permisos que herdará o grupo propietario: ningún
default:group:g-asi-profes:r--   -- Entrada que herdarán obxectos secundarios futuros: g-asi-profes:
                                 -- PERO OLLO: aquí só aparecen as entradas herdables, pero polo feit
                                   son permisos que ten a carpeta principal. Para asignar os mesmos pe
                                   group:g-asi-profes:r--


default:mask::r--
default:other::---                     -- Permisos que herdará a clase outros: ningún
```

Observar:

- Que o grupo g-asi-profes aparece dúas veces, unha para os permisos da propia carpeta **asi** e outra para que os permisos sexan propagados ás subcarpetas e arquivos futuros que se creen dentro de **asi**.
- Pola contra, o grupo g-asi-alum só ten permisos de lectura en asi e non aparece a maiores nas entradas **default**, por tanto, esa entrada non será propagada ós obxectos secundarios que se creen dentro de **asi**.

Comandos que se usaran nesta sección

Os dous primeiros son moi semellantes, pero o primeiro ademais ten a opción de usar un entorno de traballo interactivo.

chown

- **Descrición**: permite cambiar o usuario e grupo propietario dun ficheiro ou carpeta.

(change owner)

- **Sintaxe**:

```
Emprego: chown [OPCIÓN]... [DONO][:[GRUPO]]... FICHEIRO...
      O DONO e o GRUPO poden ser numéricos ou simbólicos.

 Opcións máis comúns
  -R, --recursive: opera sobre ficheiros e directorios recursivamente .
```

```
Exemplos:
  chown root /u                 Muda o dono de /u para "root".
  chown root:persoal /u         Igualmente, mais muda tamén o seu grupo para "persoal".
  chown -R root /u      Muda o dono de /u e os ficheiros e subcarpetas para "root".
```

chmod

- **Descrición**: permite cambiar os permisos dunha carpeta ou ficheiro usuario e grupo propietario dun ficheiro ou carpeta. (change mod)

- **IMPORTANTE**: este comando só o pode executar o usuario root (directamente ou con *sudo chmod*) ou o usuario propietario da carpeta ou ficheiro.

- **Sintaxe**:

```
Emprego: chmod [OPCIÓNS]... permisos ... FICHEIRO/CARPETA...

 Opcións máis comúns
  -R, --recursive: opera sobre ficheiros e directorios recursivamente .

Exemplos:
  chmod 750 /u          Sobre a carpeta /u o usuario ten permisos de (rwx), o grupo (r-x) e os demais
```

getfacl

- **Descrición**: amosa a lista de control de acceso dunha carpeta ou ficheiro. Este comando ben co paquete acl.

- **Sintaxe**:

```
Emprego: getfacl [OPCIÓNS]... FICHEIRO/CARPETA

 Opcións máis comúns
  -R, --recursive: opera sobre ficheiros e directorios recursivamente .
  -d,  amosa só as entradas herdables da lista de control de acceso.

Exemplos:
  getfacl -R /u         Amosa os permisos básicos e estendidos do directorio /u e do seu contido rec
```

setfacl

- **Descrición**: introduce, elimina ou modifica entradas da lista de control de acceso.

- **Sintaxe**:

```
Emprego: setfacl [OPCIÓNS] usuario/grupo:permisos FICHEIRO/CARPETA

 Opcións máis comúns
  -b —- borra tódalas entradas das acl.
  -k —- borra tódalas entradas herdables da acl.
  -d  -- modificador que afecta ás entradas herdables.
  -R, --recursive: opera sobre ficheiros e directorios recursivamente .

  -m, --modifica a acl.
  -x,  -- elimina unha entrada da acl.

  Usuario: u:usuario
  grupo: g:grupo

  permisos: r | w | x

Exemplos:
  setfacl -Rm g:users:rx /u    Introduce na acl de /u de tódolos
                               seus subdirectorios e ficheiros permisos
                               de lectura e escritura para o grupo users.
                               Que o faga recursivo só afecta ás carpetas/ficheiros actuais e non ás

  setfacl -Rdm g:users:rx /u   Igual que caso anterior, pero agora cando
                               se cree unha carpeta/ficheiro dentro de /u
                               tamén vai herdar a entrada da acl onde se
                               indica que os membros do grupo users podes ler e executar.

  setfacl -dx g:users /u       Borra a entrada anterior da acl.
```

Instalación e configuración ACLs

Instalar o paquete **acl** en server00.

```
sudo apt-get install acl
```

Agora queda activar en /etc/fstab en que sistemas de arquivos se quere que estean activas ás ACLs, vaise escoller en /home e raíz /. Editar o ficheiro **/etc/fstab** e engadir o prámetro **acl** ó punto de montaxe **/** e **/home**. **Ollo:** Non copiar as seguintes liñas!!!! Cada quen ten un UUID ou un dispositivo de disco/partición distinto.

```
# / was on /dev/sda1 during installation
UUID=9663c1de-e87f-4ba7-b56f-9f35b4061643 /               ext4    errors=remount-ro,acl 0       1
```

```
# /home was on /dev/sda6 during installation
UUID=e162c2ac-dbc6-446b-9adf-09040428eb97 /home/iescalquera    ext4    defaults,acl        0       2
```

Para facer efectivo o cambio sen reiniciar o servidor: volver a remontar o sistema de arquivos de **/** e **/home** para que collan o novo parámetro (acl).

```
sudo mount / -o remount
sudo mount /home/iescalquera -o remount
```

Axuste de permisos

Os únicos usuarios que poden cambiar os permisos dunha carpeta ou dun ficheiro son o usuario propietario do obxecto ou o usuario root (directamente ou a a través de sudo <comando>). Vista a introdución e os comandos *chmod*, *getfacl* e *setfacl* enriba explicados, vaise estudar con distintos exemplos a inserción, modificación e borrado de acls así como a propagación de permisos.

No servidor e co usuario administrador:

- Crear a carpeta **/comun**

```
administrador@server00:~$ sudo mkdir /comun
administrador@server00:~$ ls / -l

...
drwxr-xr-x   2 root     root      4096 2010-03-15 17:01 comun
...
```

Observar que o usuario e grupo propietario é root e os permisos son root:rwx, root(grupo):r-x e outros r-x.

- Obter a acl de */comun*, observar a información anterior disposta doutra forma.

```
administrador@server00:~$ getfacl /comun
getfacl: Removing leading '/' from absolute path names
# file: comun
# owner: root
# group: root
user::rwx
group::r-x
other::r-x
```

- Cambiar os permisos de */comun* para que a clase **outros** non teña ningún

permiso e así afinar o que se desexe con acls.

```
administrador@server00:~$ sudo chmod 750 /comun
administrador@server00:~$ ls / -l
...
drwxr-x---   2 root     root      4096 2010-03-15 17:01 comun
...
```

- Obter a acl de */comun* unha vez cambiados os permisos, observar a información anterior disposta doutra forma. Fixarse como a clase **outros** non ten ningún permiso.

```
administrador@server00:~$ getfacl /comun
getfacl: Removing leading '/' from absolute path names
# file: comun
# owner: root
# group: root
user::rwx
group::r-x
other::---
```

- Permitir ó usuario **alberto** que poida ler e acceder (r-x) á carpeta */comun*, só a esa carpeta. (-m)

```
administrador@server00:~$ sudo setfacl -m u:alberto:rx /comun

administrador@server00:~$ getfacl /comun
getfacl: Removing leading '/' from absolute path names
# file: comun
# owner: root
# group: root
user::rwx
user:alberto:r-x
group::r-x
mask::r-x
other::---
```

Observar como aparece o usuario **alberto** con permisos r-x. A outra entrada user::rwx, refírese ó usuario propietario.

- Listar carpetas con acls: Observar o caracter **+**, indica que esa carpeta ten unha acl.

```
administrador@server00:~$ ls / -l
...
drwxr-x---+  2 root     root      4096 2010-03-15 17:01 comun
```

```
drwxr-x---  2 root root 4096 2010-03-15 17:01 comun
...
```

- Crear unha subcarpeta en */comun*: **/comun/oficina**. E obter as acls de /comun e subcarpetas (-R).

```
administrador@server00:~$ sudo mkdir /comun/oficina

administrador@server00:~$ sudo getfacl -R /comun
getfacl: Removing leading '/' from absolute path names
# file: comun
# owner: root
# group: root
user::rwx
user:alberto:r-x
group::r-x
mask::r-x
other::---

# file: comun/oficina
# owner: root
# group: root
user::rwx
group::r-x
mask::r-x
other::r-x
```

Observar como ó crear a carpeta **oficina** non se herdou **alberto** de **comun**.

- Engadir unha acl para **felipe** con permisos (rwx), que se propaguen polas subcarpetas e arquivos existentes en */comun*. (-R)

```
administrador@server00:~$ sudo setfacl -Rm u:felipe:rwx /comun

administrador@server00:~$ sudo getfacl -R /comun
getfacl: Removing leading '/' from absolute path names
# file: comun
# owner: root
# group: root
user::rwx
user:alberto:r-x
user:felipe:rwx
group::r-x
mask::rwx
other::---

# file: comun/oficina
# owner: root
# group: root
user::rwx
user:felipe:rwx
group::r-x
mask::rwx
other::r-x
```

```
other::r-x
```

Observar como se obteu a lista de acls facendo uso de *sudo*, porque o usuario **administrador** non ten permisos para ler dentro de */comun*. O usuario **felipe:rwx** está na acl de comun e na da subcarpeta **oficina**.

- Finalmente engadir unha entrada á acl, usuario **xan:rwx**, que sexa herdable, de xeito que cando no futuro se cre un ficheiro ou subcarpeta-ficheiro en */comun* herde esa entrada automaticamente (-d).

```
administrador@server00:~$ sudo setfacl -dm u:xan:rwx /comun
administrador@server00:~$ sudo mkdir /comun/finanzas
administrador@server00:~$ sudo getfacl -R /comun

getfacl: Removing leading '/' from absolute path names
# file: comun
# owner: root
# group: root
user::rwx
user:alberto:r-x
user:felipe:rwx
group::r-x
mask::rwx
other::---
default:user::rwx
default:user:xan:rwx
default:group::r-x
default:mask::rwx
default:other::---

# file: comun/oficina
# owner: root
# group: root
user::rwx
user:felipe:rwx
group::r-x
mask::rwx
other::r-x

# file: comun/finanzas
# owner: root
# group: root
user::rwx
user:xan:rwx
group::r-x
mask::rwx
other::---
default:user::rwx
default:user:xan:rwx
default:group::r-x
default:mask::rwx
default:other::---
```

- Observar:
  - **o usuario xan**: non ten permisos sobre **/comun** pois só está como herdable para futuros obxectos secundarios (neste caso *finanzas*). Co

cal **xan** ten permisos en */comun/finanzas*, pero non os ten en **/comun**, por tanto, non vai poder acceder a ningunha das dúas carpetas.
- **o usuario xan**: non está en **/comun/oficina**, pero si na carpeta **finanzas** que se creou despois de introducir a acl en */comun*. Default indica que esa entrada é herdable. **xan** non aparece en oficina, porque esa subcarpeta xa estaba creada antes de meter a entrada na acl.

- Para que **xan** poida acceder a carpeta **/comun** é preciso engadir unha nova entrada:

```
administrador@server00:~$ sudo setfacl -m u:xan:rwx /comun

administrador@server00:~$ sudo getfacl /comun
 getfacl: Removing leading '/' from absolute path names
# file: comun
# owner: root
# group: root
user::rwx
user:alberto:r-x
user:felipe:rwx
user:xan:rwx
group::r-x
mask::rwx
other::---
default:user::rwx
default:user:xan:rwx
default:group::r-x
default:mask::rwx
default:other::---
```

Agora si, que **xan** pode acceder á */comun* é as subcarpetas nas que teña permisos.

- Como borrar unha acl?: eliminarase ó usuario **felipe** da acl. Primeiro de */comun* e logo recursivamente de tódalas subcarpetas. (-x)

```
administrador@server00:~$ sudo setfacl -x u:felipe /comun

administrador@server00:~$ sudo getfacl -R /comun
getfacl: Removing leading '/' from absolute path names
# file: comun
# owner: root
# group: root
user::rwx
user:alberto:r-x
user:xan:rwx
group::r-x
mask::r-x
other::---
default:user::rwx
default:user:xan:rwx
default:group::r-x
```

```
default:group::r-x
default:mask::rwx
default:other::---

# file: comun/oficina
# owner: root
# group: root
user::rwx
user:felipe:rwx
group::r-x
mask::rwx
other::r-x

# file: comun/finanzas
# owner: root
# group: root
user::rwx
user:xan:rwx
group::r-x
mask::rwx
other::---
default:user::rwx
default:user:xan:rwx
default:group::r-x
default:mask::rwx
default:other::---
```

Observar que **felipe** está eliminado da acl de */comun* pero non da subcarpeta *oficina*.

- Para eliminalo de */comun* e de tódalas súas subcarpetas e arquivos hai que facelo recursivamente: (-Rx)

```
administrador@server00:~$ sudo setfacl -Rx u:felipe /comun

administrador@server00:~$ sudo getfacl -R /comun
getfacl: Removing leading '/' from absolute path names
# file: comun
# owner: root
# group: root
user::rwx
user:alberto:r-x
user:xan:rwx
group::r-x
mask::r-x
other::---
default:user::rwx
default:user:xan:rwx
default:group::r-x
default:mask::rwx
default:other::---

# file: comun/oficina
# owner: root
# group: root
user::rwx
group::r-x
mask::r-x
other::r-x
```

```
# file: comun/finanzas
# owner: root
# group: root
user::rwx
user:xan:rwx
group::r-x
mask::rwx
other::---
default:user::rwx
default:user:xan:rwx
default:group::r-x
default:mask::rwx
default:other::---
```

- Á vista dos exemplos anteriores:
  - Para traballar con grupos no canto de usuarios, usar: **g:grupo**
  - Non confundir recursividade (-R) con herdanza (-d). O primeiro afecta a carpeta e subcarpetas/arquivos existentes, o segundo non afecta á carpeta senón ás subcarpetas e arquivos que se creen nun futuro.
  - Para borrar tódalas acls, habería que usar o parámetro **-b**.
  - Para modificar os permisos do usuario propietario, pódese usar chmod ou: **setfacl -opcións u::permisos /comun**. Para o grupo propietario sería semellante pero g:: no canto de u::.

Uso de ACLs a través de NFS

Se se exporta unha carpeta con ACLs a través de NFS, ó proceso é transparente nas actuais versións de NFS.

Antes de continuar no cliente débese instalar o paquete acl do mesmo xeito que se fixo do servidor:

```
sudo apt-get install acl
```

Exportar /comun

Engadir en server00 ó ficheiro **/etc/exports** unha nova exportación.

```
# /etc/exports
/comun  *(rw,async)
# O * indica que a exportación é para calquera rede IP.
```

Facer efectiva a exportación:

```
sudo exportfs -ra
```

Montar /comun no cliente

Crear no cliente **/mnt/comun**

```
sudo mkdir /mnt/comun
```

Editar o ficheiro /etc/fstab do cliente e engadir o punto de montaxe NFS:

```
#/etc/fstab do cliente
10.0.0.100:/comun /mnt/comun nfs defaults,vers=3        0       0
```

Montar os puntos de montaxe de /etc/fstab que non estean activos:

```
sudo mount -a
```

- Comprobar as ACLs exportadas.

```
administrador@cliente00:~$ getfacl -R /mnt/comun

getfacl: Removing leading '/' from absolute path names
# file: mnt/comun
# owner: root
# group: root
user::rwx
user:root:r-x
user:alfredo:r-x
user:xan:rwx
group::r-x
mask::rwx
other::---
default:user::rwx
default:user:xan:rwx
default:group::r-x
default:mask::rwx
default:other::---

getfacl: /mnt/comun: Permission denied
```

Observar como ó usuario administrador do cliente non lle deixa acceder as ACLs das subcarpetas de */mnt/comun*. Iso é porque o usuario administrador do cliente caería dentro da clase **outros**, que xustamente non ten ningún permiso na ACL de */mnt/comun*.

Xestionar as ACLs no cliente

Lembrar:

- que os permisos só poden ser cambiados polo usuario propietario ou polo usuario root.
- as exportacións no server fixéronse **Confiando en todo o mundo excepto root**: Desa forma o servidor NFS mapeará o UID do usuario do equipo cliente cun usuario do servidor excepto se se trata do usuario root, que será mapeado como usuario anónimo.

Por tanto no cliente só poden xestionar os permisos dunha exportación os usuarios propietarios do server que non sexan root.

Entón tense:

- No exemplo que se ten co punto de montaxe: **/mnt/comun** O usuario propietario dese **comun** é o root do servidor non o root do cliente, por tanto o usuario root do cliente non vai poder cambiar ningún permiso, nin tomar posesión, cambiar o propietario, etc, das carpetas e subcarpetas exportadas por un servidor.
- Os usuarios do servidor que sexan donos de carpetas, subcarpetas ou ficheiros exportadas si van poder modificar os permisos de aquelas nas que son donos.
- Se o usuario root desexa facer cambios, debe ser o root no servidor, ben a través de **ssh** ou directamente no servidor, ou facendo uso de **sudo**, pero sempre no servidor.

Xestión gráfica de ACLs: eiciel

Todo canto se fixo anteriormente pode ser realizado cunha ferramenta gráfica chamada Eiciel creada por un Catalán Roger Ferrer .

Esta ferramenta precisa dun entorno gráfico, por tanto, só se pode instalar nos clientes, e facilítalle ós usuarios *normais* a xestión de permisos, pois engade as solapas de propiedades dunha carpeta ou un ficheiro, unha nova lapela para xestionar graficamente a lista de control de acceso.

Eiciel, o que fai é traballar directamente coas acls, co cal estas poden ser modificadas tanto por consola como graficamente.

- Instalar Eiciel no cliente:

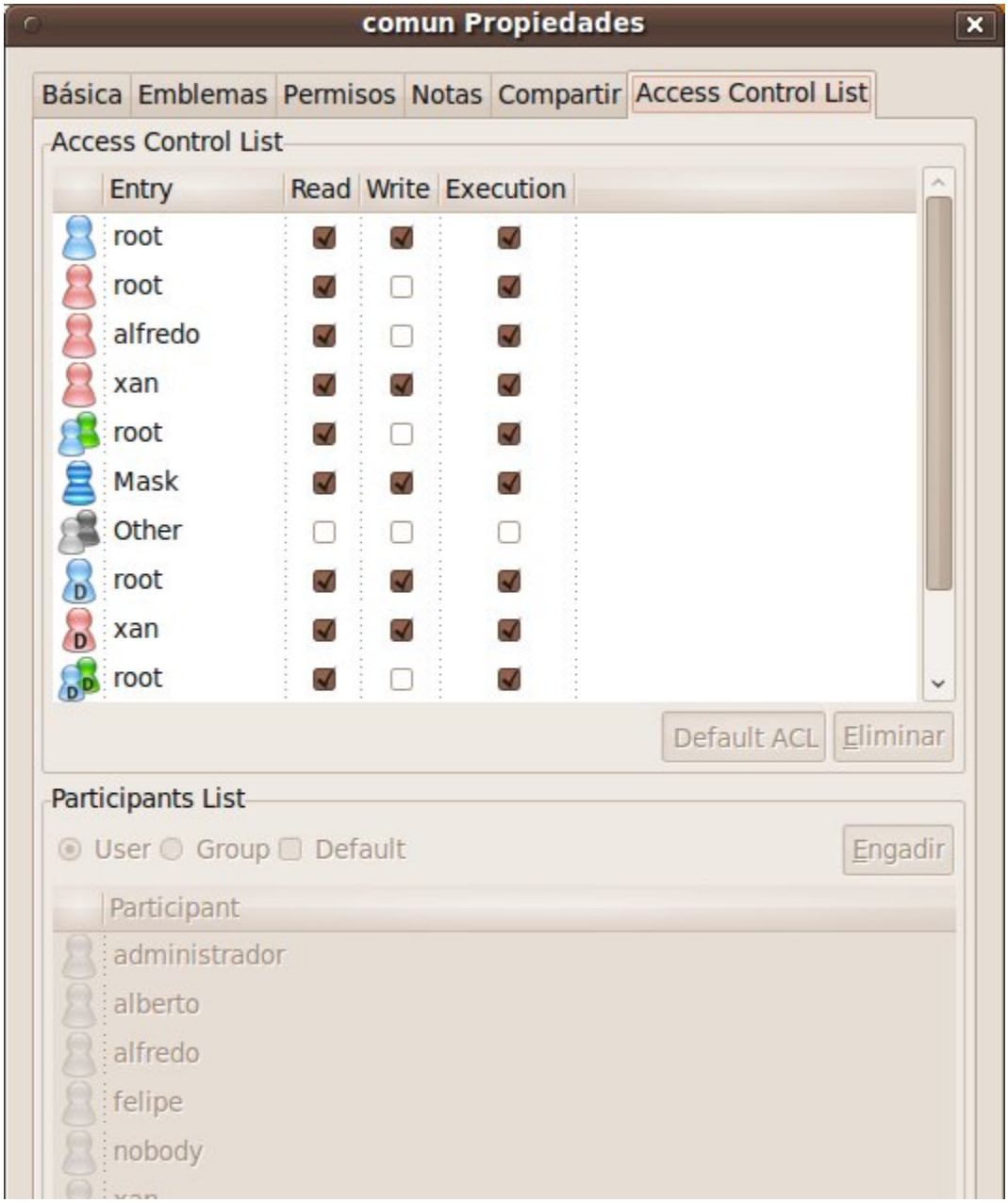
```
sudo apt-get install eiciel
```

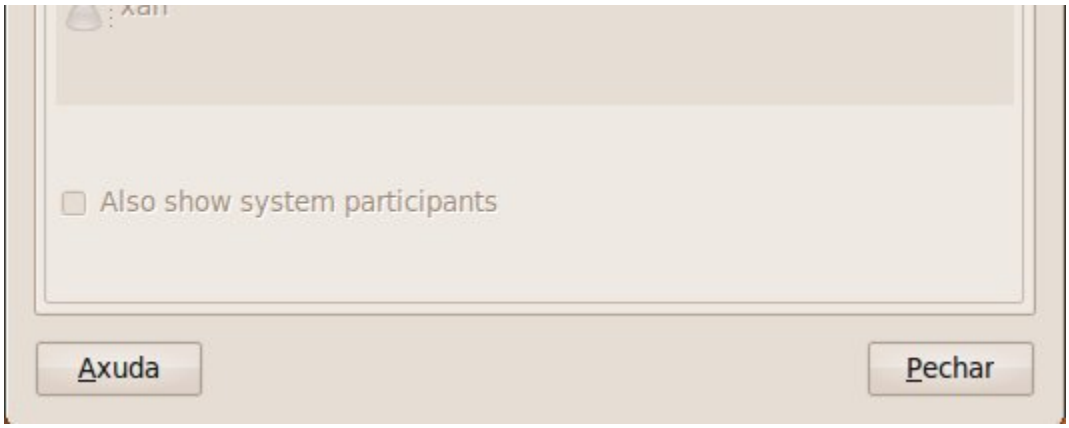
- No menú do cliente: **Aplicativos->Ferramentas do sistema** hai unha entrada para Eiciel.

Pero eiciel nin as acls poden ser usados en ningún dos sistema de arquivos físicos do cliente, pois no */etc/fstab* do cliente non está o parámetro acl para ningún dos puntos de montaxe.

Para os puntos de montaxe nfs, si se poden usar as acls/eiciel pois as listas de control de acceso son transportadas por nfs de xeito transparente.

- Co usuario **administrador** do cliente usando nautilus ir a propiedades da carpeta **comun** que está en **/mnt** (**Ollo:** Despois de instalar o paquete eiciel, teremos que reiniciar a sesión para que a pestana de ACLs apareza).

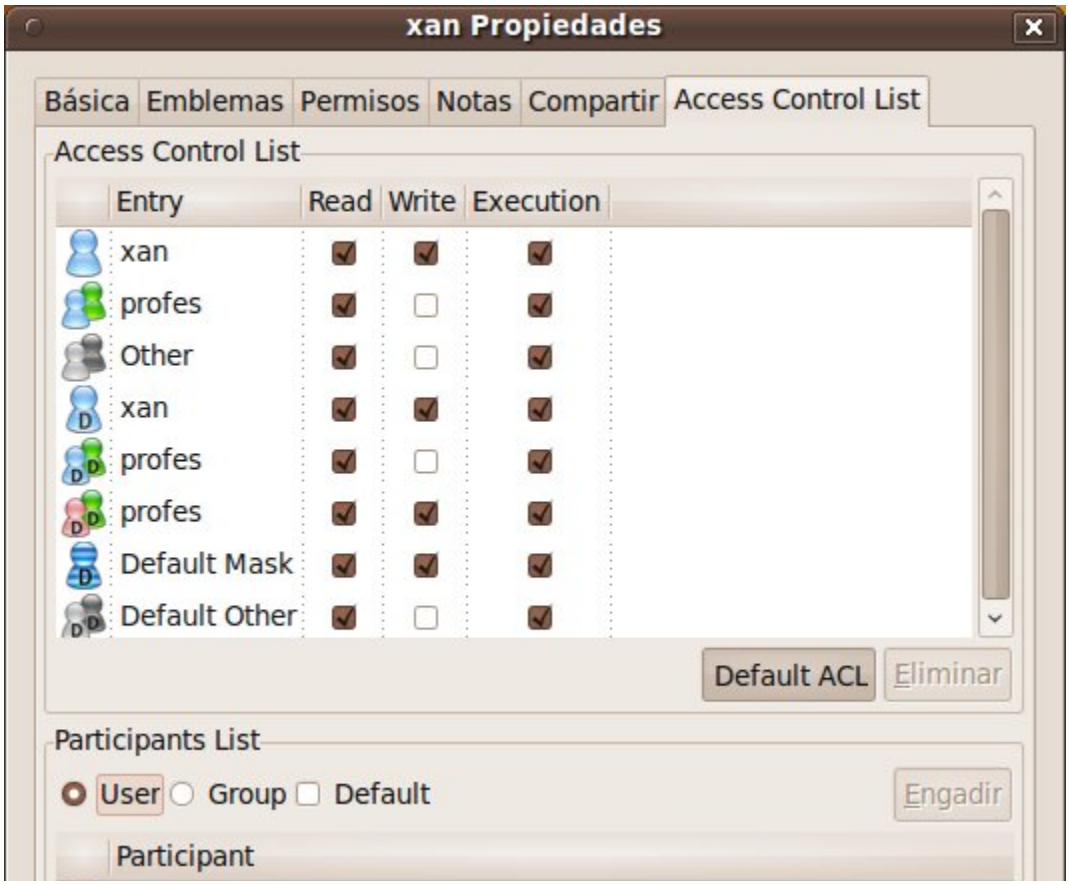


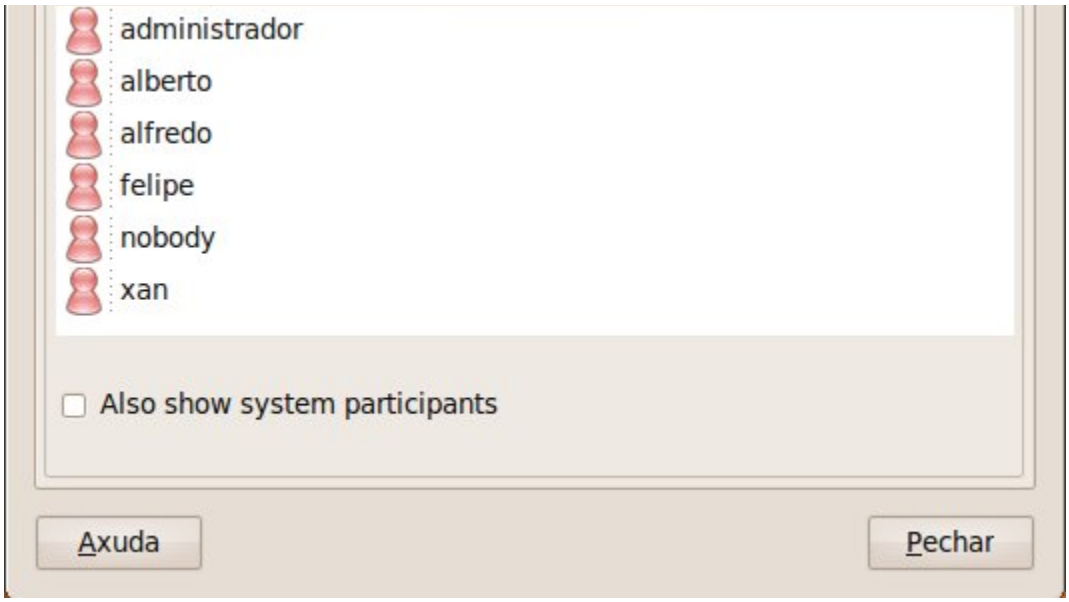


Observar como o usuario **administrador** do cliente só pode ver pero non cambiar nada.

Para familiarizarse coa interpretación das iconas, cores e formatos véxase a seguinte táboa 1 do enlace: http://rofi.roger-ferrer.org/eiciel/doc/ar01s02.html#manipular-acl

- Pechar a sesión co cliente e agora entrar co usuario **xan** en contorno gráfico. Ir as propiedades da súa carpeta persoal.





Observar como agora xan pode engadir máis usuario, grupos, poñer usuarios e grupos por defecto (herdables), etc.

Agora **xan** pode ir a comun (/mnt/comun) do cliente e crear obxectos.

-- Antonio de Andrés Lema e Carlos Carrión Álvarez