IES de Teis
Avenida. de Galicia, 101
36216 – Vigo

Tfno.: 886 12 04 64
e-mail: ies.teis@edu.xunta.es
http://www.iesteis.es

Permisos Linux

Los permisos sobre recursos de Linux se remontan a los Unix de los años 70 del siglo pasado. En ese momento, los equipos informáticos eran fundamentalmente sistemas centralizados Unix que funcionaban como servidores de terminales remotos. Como ejemplos de estos sistemas podríamos nombrar los DEC PDC-11 o los IBM System/360.

En estos servidores se ejecutaban distintos programas o servicios y era necesario establecer algún tipo de seguridad. Era necesario desarrollar algún sistema de seguridad que no supusiera demasiada sobrecarga y que fuera sencillo de entender y aplicar. Se decidió usar permisos a tres niveles:

- Propietario
- Grupo
- Otros

El propietario es el creador del recurso y su papel debía ser el de gestor más importante de este recurso. No era un concepto desconocido en ese momento ni ahora, de hecho, cuando instalamos una base de datos, una de las preguntas de debemos contestar durante la instalación es la contraseña del usuario "root" de la base de datos (no confundir con el root del sistema operativo).

Pongamos como ejemplo que instalamos "Apache", el conocido servidor web. Llamaremos a nuestro usuario propietario "apache" y será el único que tiene permisos para lanzar el servidor con "/usr/bin/apache2 start", configurarlo en "/etc/apache2/apache2.conf", pararlo, comprobar el estado, etc.

Este usuario propietario no necesariamente se tiene que corresponder a una persona física. En este caso el usuario "apache" simplemente es el que tiene la capacidad de arrancar y parar el servidor web.

Por otro lado tenemos los grupos. Habitualmente junto a la creación de un usuario se crea también un grupo con el mismo nombre y se relacionan uno como miembro del otro. Esto es apropiado para estos casos puesto que podríamos hacer que hubiera un grupo de usuarios con permisos restringidos con el servidor apache de manera que, aunque no puedan encenderlo y apagarlo sí puedan configurarlo y añadir contenido. En ese caso añadiremos al grupo "apache" todos los usuarios a los que queremos conceder permisos de edición.

Por último está el resto que son todos los otros usuarios que no pueden arrancar el servidor, ni editarlo pero sí pueden leer las páginas web.

En el ejemplo del servidor web, los tres niveles interactúan con el recurso pero esto no siempre tiene porqué ser así. Pensemos por ejemplo en el servicio de impresión. En este caso el usuario propietario "impresora" tiene el control total y puede usar y configurar la impresora. Los miembros del grupo

"impresora" podrán solamente usarla para imprimir. El resto de los usuarios no tendrá ningún permiso sobre este recurso.

Otro ejemplo mucho más estricto puede ser al acceso a una carpeta o archivo al que solo queremos que acceda el "usuarioX". En este caso tendremos que dar la propiedad a este usuario y darle permisos en el apartado de propietario. A nivel de grupo y otros no daremos ningún permiso. De esta manera solo el "usuarioX" podrá acceder a la carpeta y ni sus colaboradores ni el resto podrá acceder.

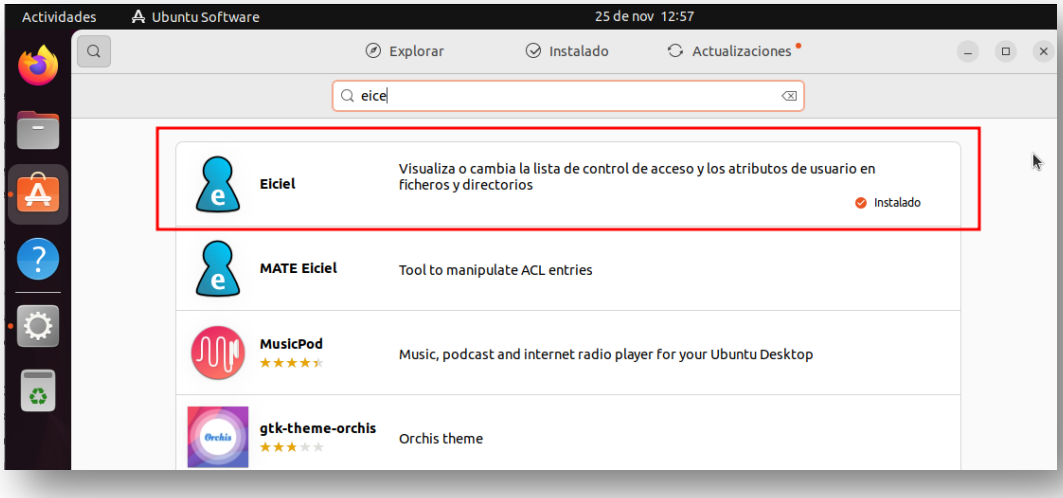
En el momento de la creación del recurso se le asigna como usuario y grupo propietario el del usuario creador. Esto se puede cambiar de manera que podemos cambiar el usuario propietario, el grupo propietario o ambos haciendo que las posibilidades de asignación de permisos sean mucho más amplias.

La llegada de los permisos Windows en la década de los 90 del siglo pasado dejó algo anticuado al sistema de permisos Unix pero rápidamente se adaptaron con la inclusión de las ACLs y algunos permisos especiales como por ejemplo el "sticky bit" (solo puede borrar el propietario).

Al igual que nos pasaba en la guía de gestión de usuarios y grupos locales en Linux, en Ubuntu no disponemos de una aplicación gráfica por defecto para la gestión de permisos que incluya de manera nativa las ACLs y será necesario instalar alguna.

Puedes elegir cualquiera de las que están disponibles en los repositorios. En esta guía usaremos "Eiciel" porque además de ser una aplicación gráfica también es un plugin para "Nautilus" (el explorador de ficheros por defecto en Ubuntu) de manera que podremos tener una experiencia gráfica desde el propio navegador al igual que con Windows.

Instalar "Eiciel" es tan sencillo como instalarlo desde la tienda de aplicaciones de Ubuntu.



Políticas de seguridad

En Linux tenemos permisos muy simples consistentes en:

- Permiso de lectura (r)
- Permiso de escritura (w)
- Permiso de ejecución (x)

El permiso de lectura aplicado a ficheros permite ver su contenido y aplicado a carpetas permite listar los archivos y subcarpetas contenidos en ella.

El permiso de escritura permite la edición de un archivo para modificar su contenido pudiendo ampliar o reducir el mismo. La clave es que seguirá existiendo el archivo, no se puede borrar, podemos alterar su contenido pero no su existencia. El permiso de escritura aplicado a una capeta permite crear, renombrar y eliminar archivos. La diferencia es que se puede alterar su existencia pero no su contenido.

El permiso de ejecución aplicado a un archivo permite que ese archivo sea tratado como un script o archivo binario y que se pueda lanzar directamente como un programa. El permiso de ejecución aplicado a una carpeta permite el acceso a la misma y la capacidad de listar los archivos y subcarpetas allí contenidos.

En Linux siempre tendremos permisos a nivel de usuario propietario, grupo propietario y otros. Por ello es necesario tener en cuenta este último grupo de "Otros" porque será la ACL que se aplique a cualquier usuario que no esté explícitamente indicado con anterioridad.

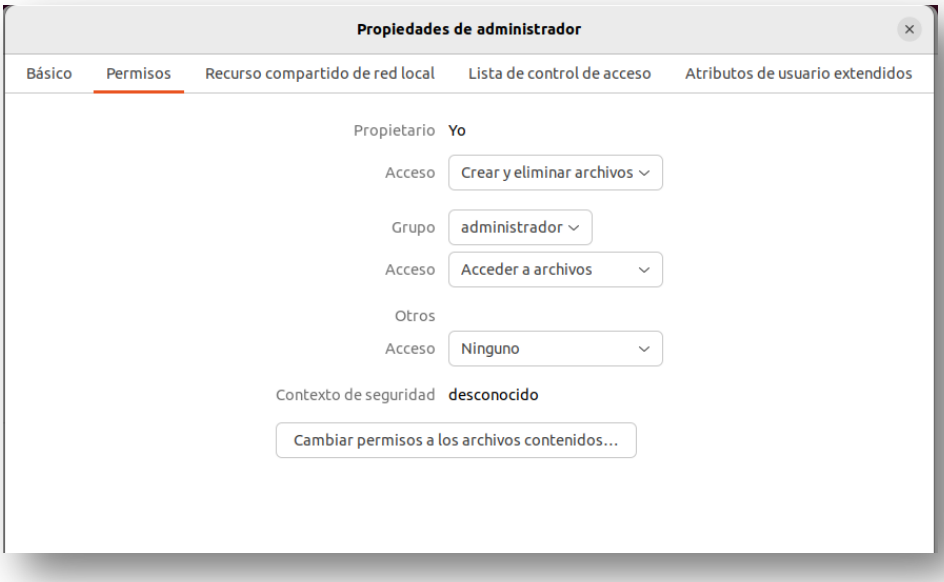
En Linux los permisos se aplican en un orden concreto. En primer lugar se aplicarán los permisos clásicos de Unix buscando la aplicación de permisos según coincida el propietario y grupo. En caso de quei un usuario o grupo tenga acceso basado en esos permisos, se otorga. Después se evalúan y aplican las ACLs. Los permisos más restrictivos prevalecerán incluso sobre los tradicionales. Si las ACLs tienen permisos más permisivos se combinarán los provenientes de los tradicionales y las ACLs.

Hay que tener en cuenta que en Linux, el usuario propietario del archivo tiene permisos completos por defecto incluso si se definen ACLS específicas.

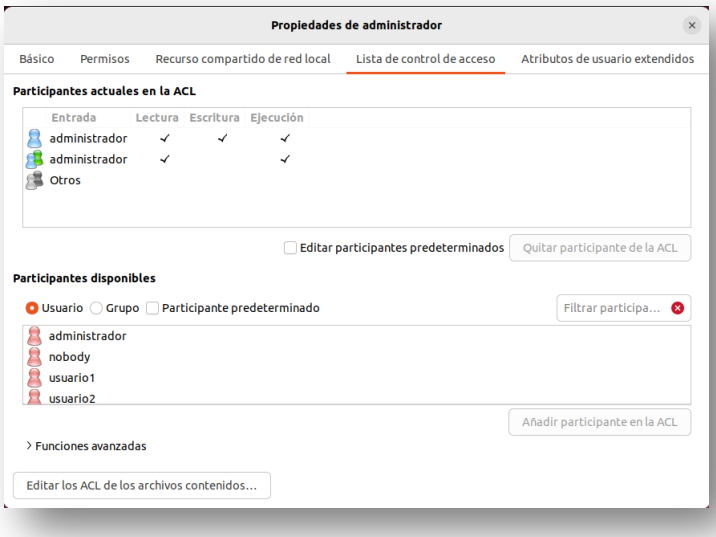
Consultar permisos (GUI)

La consulta de permisos a una carpeta o archivo desde una vista gráfica podemos hacerla desde el propio explorador de ficheros "Nautilus". Para ello seleccionamos el recurso y pinchamos sobre él con el botón derecho para llegar hasta sus propiedades.

Se abrirá una ventana y seleccionaremos la pestaña "Permisos". Desde allí podremos ver quién es el propietario y qué permisos tiene. Lo mismo para el grupo y el resto de los usuarios. El "Contexto de seguridad" tiene que ver con una política de permisos más avanzada "SELinux" que no veremos en esta guía. Por último, el botón de "Cambiar permisos a los archivos contenidos" hace referencia a la posibilidad de cambiar de manera recursiva los permisos de los recursos internos a esta carpeta.



Al haber instalado "Eiciel" disponemos otras maneras más potentes de consultar gráficamente los permisos. Para ello abriremos igualmente las propiedades del archivo o carpeta que queramos consultar. Una vez abierta la ventana de propiedades pincharemos sobre la pestaña "Lista de control de acceso" y desde allí veremos en forma de ACLs cada una de las reglas que tiene el recurso.

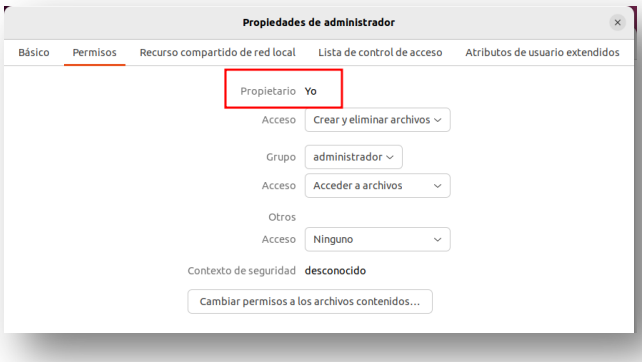


Por herencia del antiguo sistema de permisos siempre tendremos al menos las 3 ACLs que se corresponden con el usuario propietario, el grupo propietario y el resto de los usuarios. En la captura podemos ver una caja titulada "Participantes actuales en la ACL" donde aparece un listado de todas las ACLs aplicables al recurso del que estamos viendo las propiedades. En primer lugar tenemos al usuario "administrador" que tiene permisos totales, es decir, de lectura, escritura y ejecución. En segundo lugar tenemos al grupo "administrador" que solo tiene permisos de lectura y ejecución. Fijémonos en que podemos distinguir un usuario de un grupo por su icono. Por último tenemos al resto de los usuario que no tiene ningún permiso sobre el recurso.

Los permisos de esa captura hacen referencia a la carpeta personal del usuario "administrador" y podemos ver como solo él y los miembros de su grupo (aunque de manera más limitada) tienen permisos dentro de esa carpeta, cualquier otro usuario no puede ni entrar.

Cambiar usuario propietario (GUI)

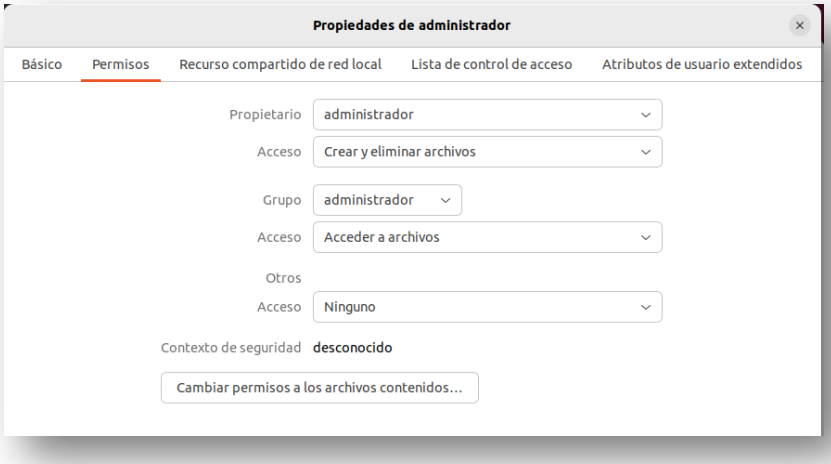
El usuario propietario de una carpeta se puede ver desde la pestaña "Permisos" de las "Propiedades" de una carpeta o fichero seleccionado.



El problema es que este cambio requiere de permisos administrativos y será necesario arrancar el explorador de ficheros "Nautilus" con permisos administrativos. La manera más rápida de hacerlo es abrir una terminal y ejecutar el siguiente comando:

```
sudo nautilus
```

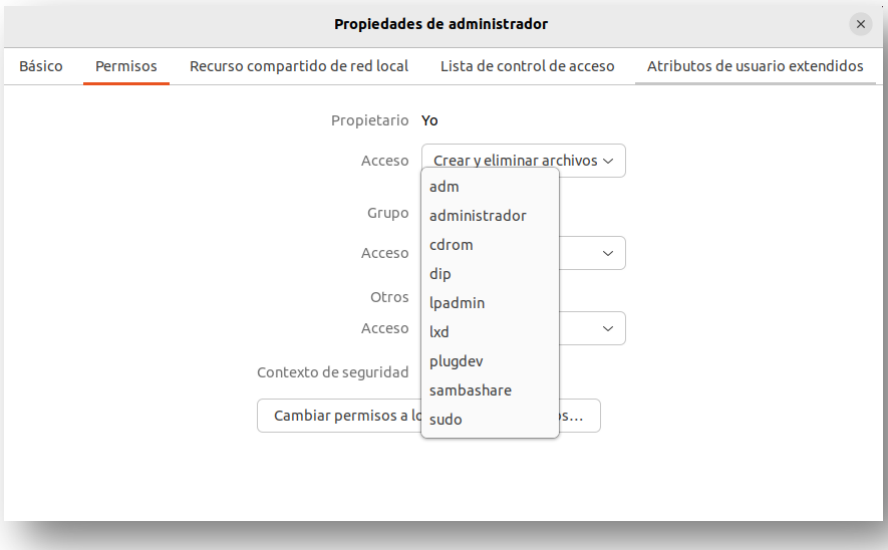
Con ello estaremos ejecutando el explorador de ficheros con permisos administrativos y tendremos acceso a todo.



Si nos fijamos en la captura anterior veremos la diferencia que hay entre haber abierto las propiedades en un explorador sin permisos como en la primera captura y haberlo abierto con permisos administrativos en esta captura. En este último caso tenemos acceso a cambiar el propietario sin más que pinchar en el desplegable y seleccionar el usuario del sistema al que queramos dar la propiedad.

Cambiar grupo propietario (GUI)

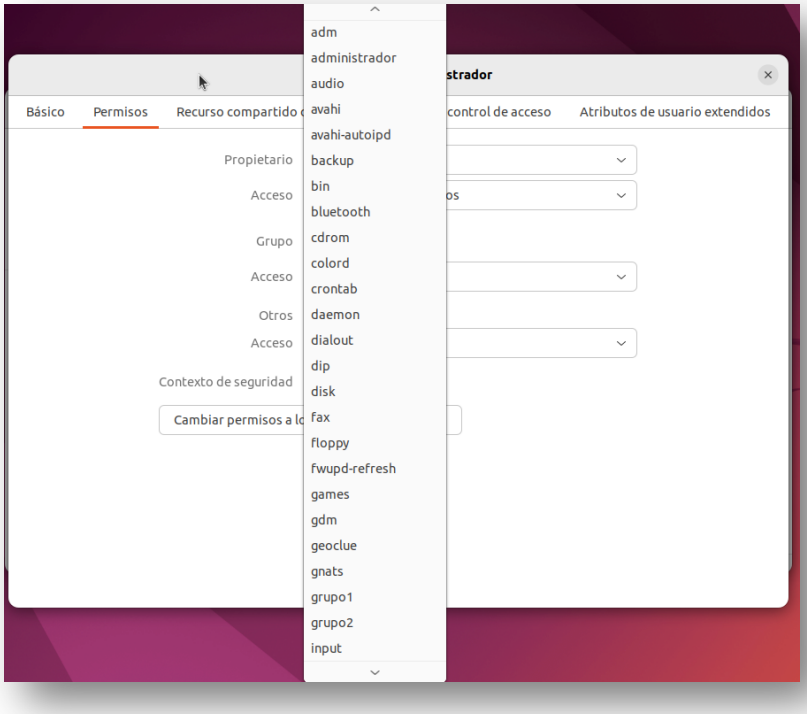
Para cambiar el grupo principal de un fichero o carpeta debemos abrir la pestaña "Permisos" de las "Propiedades" de un fichero o carpeta desde el navegador. Al lado de la etiqueta "Grupo" tendremos un desplegable con parte de los grupos locales del sistema.



Si queremos poder elegir entre todos los grupos locales del sistema tendremos que abrir el explorador de archivos "Nautilus" con permisos de administrador. La forma más sencilla de hacer esto es mediante el comando:

```
sudo nautilus
```

Una vez tengamos el explorados de ficheros abierto con permisos administrativos nos desplazamos hasta el fichero o carpeta de nuestro interés, abrimos sus "Propiedades" y pinchamos en la pestaña "Permisos".



Ahora ya sí, desde el desplegable de "Grupos" podremos seleccionar de entre todos los grupos locales.

Modificar permisos (GUI)

Un usuario puede modificar los permisos otorgados al grupo principal y otros usuarios siempre que este usuario sea el propietario del recurso. De ser así es tan sencillo como abrir las "Propiedades" del fichero o carpeta, pinchar en la pestaña "Permisos" y cambiar el los desplegables el permiso específico para el propietario, grupo u otros.

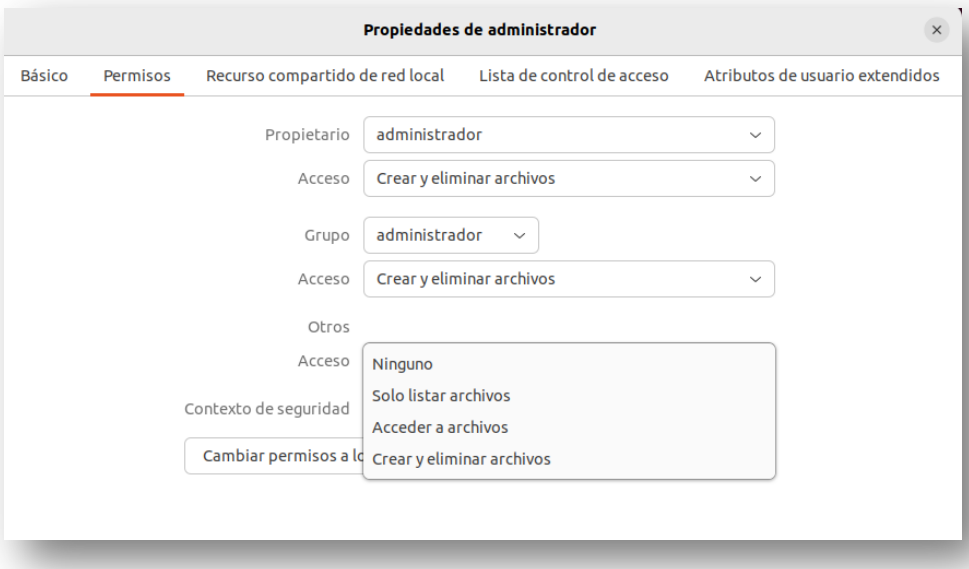
La forma de indicar los permisos no es ni simbólica ni octal, aquí la expresa en modo literal con las siguientes alternativas:

- ”Ninguno”. Es el equivalente al permiso 0 o ”- - - ”.
- ”Solo listar archivos”. Es el equivalente al permiso 4 o ”r - - ” de lectura.
- ”Acceder a archivos”. Es el equivalente al permiso 5 o ”r-x” de lectura y ejecución.
- ”Crear y eliminar archivos”. Es el equivalente al permiso 7 o ”rwx” de lectura, escritura y ejecución.

Desde el explorador solo se podrán modificar los permisos de recursos de los que somos propietarios. Si queremos modificar otros recursos debemos abrir el explorador de ficheros con permisos administrativos usando el siguiente comando:

```
sudo nautilus
```

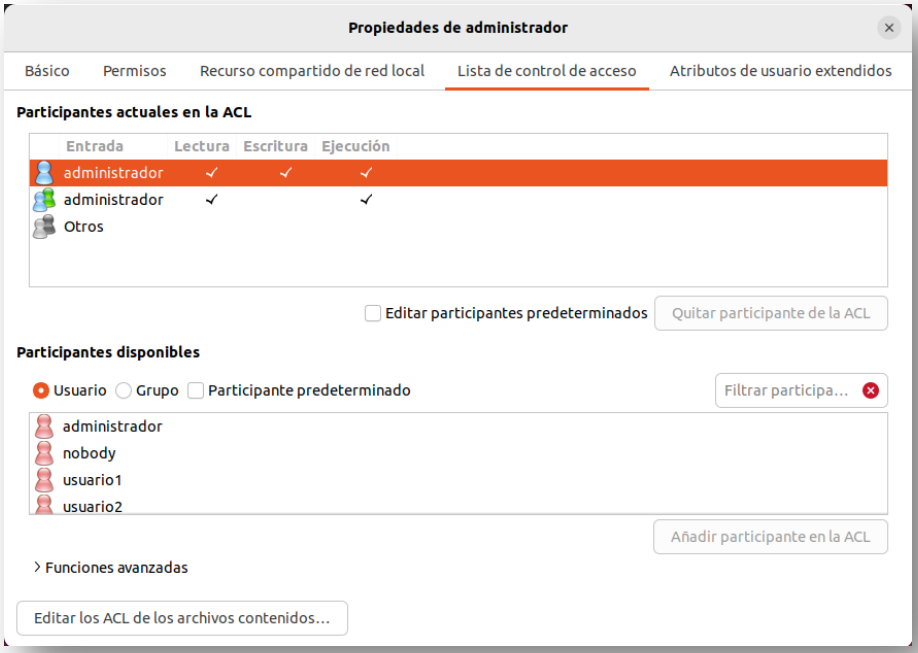
En este caso ya nos permitirá hacer cambios sin más que modificar el desplegable y cerrar la ventana de ”Propiedades”



Consultar ACLs (GUI)

Ubuntu no trae por defecto ninguna aplicación para ver gráficamente las ACLs de un determinado recurso. Al principio de la guía instalamos ”Eiciel” desde la tienda de aplicaciones que, además de la aplicación incluye un plugin para nautilus de forma que nos permitirá consultar las ACLs desde la propia ventana de propiedades del explorador de ficheros ”Nautilus”

Dentro de las ”Propiedades” del recurso seleccionado pincharemos en la pestaña ”Lista de control de acceso”. Allí veremos las ACLs actuales de ese recurso.



Siempre encontraremos las 3 ACLs heredadas de los permisos tradicionales al propietario, grupo principal y otro. Recuerda que el usuario propietario nunca va a perder acceso al recurso.

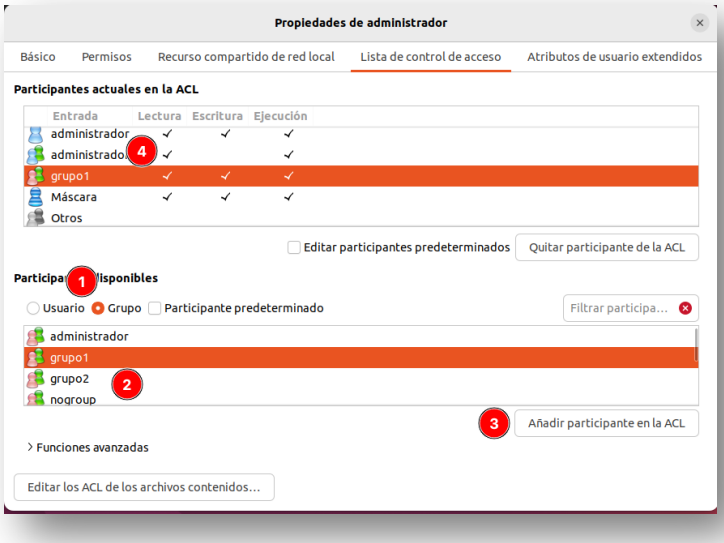
Modificar ACLs (GUI)

Modificar las ACLs requiere ser propietario del recurso o bien un usuario con permisos administrativos.

Tendremos que llegar a la pestaña "Lista de control de acceso" de las "Propiedades" del recurso en el que queramos modificar las ACLs.

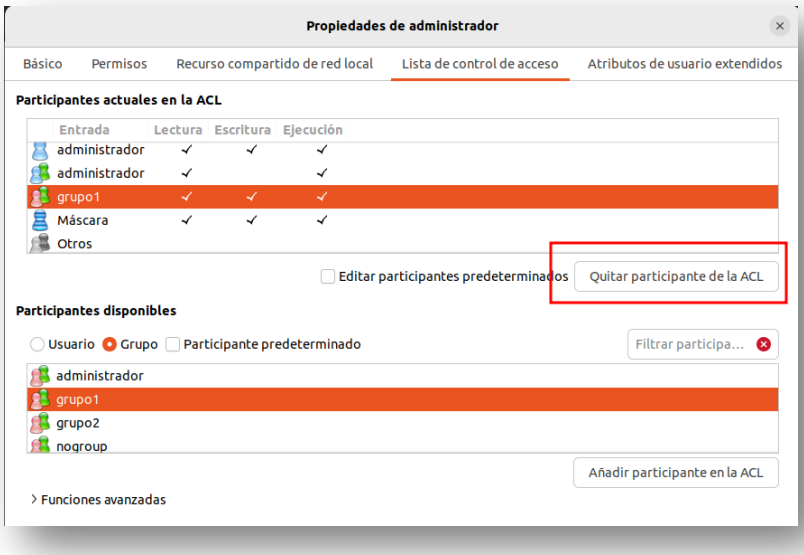
Podemos modificar alguna de las ACLs existentes sin más que pinchar en el "check" que concede el permiso de manera que, si hay check hay permiso.

También podemos añadir nuevas ACLs. En este caso, en primer lugar, en la parte inferior de esta ventana tendremos que seleccionar si la ACL se aplicará a un usuario concreto o a un grupo concreto. En función de qué opción seleccionemos veremos los usuarios o grupos locales del sistema. Seleccionaremos el que sea de nuestro interés y pulsaremos el botón "Añadir participantes en la ACL". Al hacer esto aparecerá nuestra nueva ACL en la parte superior pudiendo cambiarle los permisos como vimos anteriormente.



Al añadir nuestra primera ACL se ha creado también una ACL llamada "máscara". Esta ACL limita los permisos efectivos indicados en las ACLs. Por defecto la máscara tendrá permisos totales y será cada ACL la que determine el nivel de privilegio. Puede que en algún escenario sea interesante modificar la máscara para limitar permisos. Por ejemplo, una máscara con solo permiso "r" de lectura hará que cualquier ACL con un privilegio mayor como control total (rwx) o lectura y ejecución (r-x) se quede con los privilegios efectivos de solo lectura. Digamos que se hace una intersección entre la máscara y el permiso y solo se concede aquellos permisos que estén ambos a 1. Por supuesto esto no afecta al usuario propietario que siempre tiene control total sobre el recurso.

Por último también podemos eliminar ACLs siempre que no sean las heredadas de los permisos tradicionales. Para ello solamente tenemos que seleccionar la ACL de entre las que aparecen en la caja superior y pulsar sobre el botón "Quitar participantes de la ACL"



Consultar permisos (CLI)

Los permisos tradicionales son aquellos que nos indicaban los privilegios del usuario propietario, del grupo principal y del resto de los usuarios de un determinado recurso. Estos permisos se denotan con una letra que indica lo siguiente:

- "r" para lectura
- "w" para escritura
- "x" para ejecución.

En función de si los permisos se aplican a un fichero o carpeta tienen una semántica distinta. Por ejemplo, que un archivo tenga permiso de ejecución nos dice que puede ser tratado como una aplicación, en cambio, ese mismo permiso aplicado a una carpeta no dice que se pude entrar y listar su contenido.

La manera más sencilla para consultar los permisos tradicionales a través de comando es mediante el siguiente comando:

```
ls -la
```

El comando anterior nos listará el contenido de la carpeta actual pero podemos añadirle comodines o rutas para especificar mucho más lo que queremos ver.

```
administrador@equipo:/home$ ls -la
total 32
drwxr-xr-x  8 root          root          4096 nov 24 21:43 .
drwxr-xr-x 20 root          root          4096 nov 28  2023 ..
drwxr-x--- 16 administrador administrador 4096 nov 24 20:19 administrador
drwxr-x---  2 usuario1      usuario1      4096 nov 24 19:14 usuario1
drwxr-x---  2 usuario2      usuario2      4096 nov 23 21:48 usuario2
drwxr-x---  2 usuario3      usuario3      4096 nov 24 20:19 usuario3
drwxr-x---  2 usuario4      usuario4      4096 nov 24 20:23 usuario4
drwxr-x---  2 usuario5      usuario5      4096 nov 24 21:43 usuario5
administrador@equipo:/home$
```

La salida nos muestra en filas todo el contenido de la carpeta "/home" en este caso por se la carpeta activa en el momento de la ejecución.

La primera columna es una concatenación de información. El primer carácter nos indica el tipo de archivo que, en este caso nos indica que todos son carpeta porque todos empiezan por el carácter "d". Después viene una concatenación de 3 grupos de 3 caracteres cada uno. Los primeros tres caracteres, los de más a

la izquierda" se corresponden con los permisos del usuario propietario. Aquí podemos ver como el usuario propietario tiene control total "rwx".

Los 3 siguientes caracteres indican los permisos otorgados al grupo principal. En esta captura vemos que todos tienen privilegios limitados "r-x" consistentes en leer y ejecutar pero no permite la modificación.

Por último, los 3 caracteres de la derecha nos indican los permisos que tienen el resto de los usuarios que no son ni propietarios ni son miembros del grupo principal. En general podemos ver que solo tienen guiones indicando que no tienen concedido ese permiso.

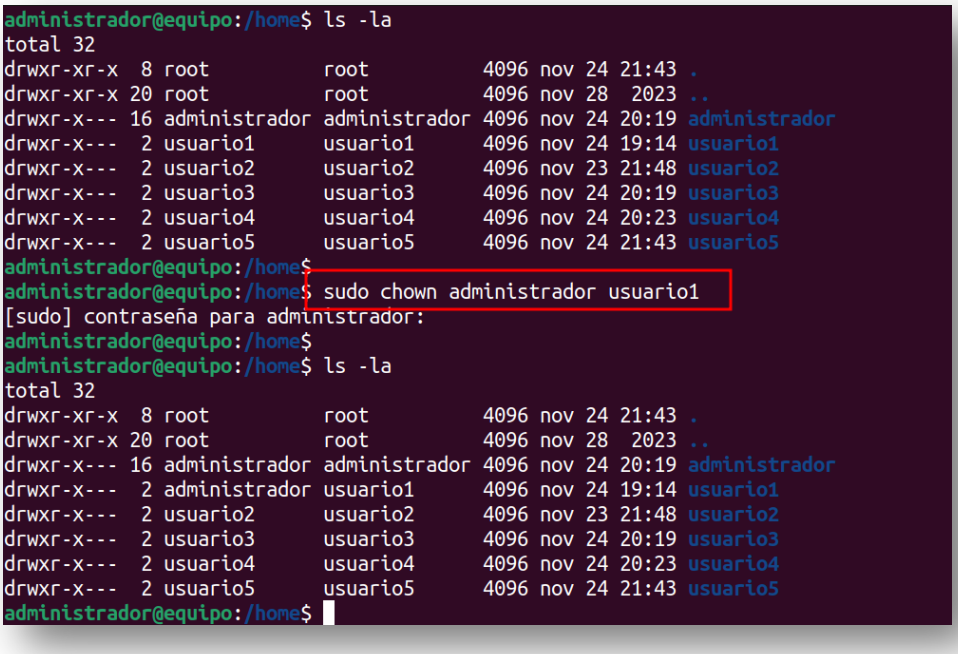
En la captura también podemos ver en la tercera y cuarta columna el usuario propietario y el grupo principal de cada recurso.

Cambiar usuario propietario (CLI)

El usuario propietario o cualquiera de los usuarios administradores puede ceder o tomar posesión de la propiedad de un archivo o carpeta. Pensemos que tener la propiedad de un recurso nos proporciona privilegios dentro de ese recurso similares a ser administrador.

Para cambiar la propiedad de un recurso ejecutaremos el siguiente comando:

```
sudo chown administrador usuario1 -R
```



El modificador "-R" hace que el comando se aplique recursivamente a todo el contenido del recurso.

En la captura anterior podemos ver como la carpeta "usuario1" tenía como propietario al "usuario1" pero que después de tomar posesión con permisos administrativos el resultado es que la carpeta "usuario1" ahora tiene a "administrador" como propietario.

Cambiar grupo propietario (CLI)

Sudo chgrp grupo1 /permisos

```
administrador@equipo:/home$ ls -la
total 32
drwxr-xr-x  8 root          root          4096 nov 24 21:43 .
drwxr-xr-x 20 root          root          4096 nov 28  2023 ..
drwxr-x--- 16 administrador administrador 4096 nov 24 20:19 administrador
drwxr-x---  2 administrador usuario1      4096 nov 24 19:14 usuario1
drwxr-x---  2 usuario2      usuario2      4096 nov 23 21:48 usuario2
drwxr-x---  2 usuario3      usuario3      4096 nov 24 20:19 usuario3
drwxr-x---  2 usuario4      usuario4      4096 nov 24 20:23 usuario4
drwxr-x---  2 usuario5      usuario5      4096 nov 24 21:43 usuario5
administrador@equipo:/home$
administrador@equipo:/home$ sudo chgrp administrador usuario1
administrador@equipo:/home$
administrador@equipo:/home$ ls -la
total 32
drwxr-xr-x  8 root          root          4096 nov 24 21:43 .
drwxr-xr-x 20 root          root          4096 nov 28  2023 ..
drwxr-x--- 16 administrador administrador 4096 nov 24 20:19 administrador
drwxr-x---  2 administrador administrador 4096 nov 24 19:14 usuario1
drwxr-x---  2 usuario2      usuario2      4096 nov 23 21:48 usuario2
drwxr-x---  2 usuario3      usuario3      4096 nov 24 20:19 usuario3
drwxr-x---  2 usuario4      usuario4      4096 nov 24 20:23 usuario4
drwxr-x---  2 usuario5      usuario5      4096 nov 24 21:43 usuario5
administrador@equipo:/home$
```

Modificar permisos (CLI)

Para modificar los permisos de un archivo o carpeta usaremos el comando chmod. A este comando tendremos que pasarle los nuevos permisos en octal o bien las modificaciones en simbólico.

Octal

Los permisos que vemos al hacer un "ls -l" son la concatenación de un carácter indicando el tipo de archivo, una terna de caracteres "rwx" indicando los permisos del usuario propietario, otra terna "rwx" indicando los permisos del grupo propietario y una última terna indicando los permisos de los otros usuarios. Cada una de estas ternas las interpretamos como un número binario donde cada una de las 3 cifras viene dada por un 1 si tiene ese permiso o un 0 so no lo tiene.

Por ejemplo, un permiso total, "rwx" sería tres unos (111) que equivale a un 7 decimal (y octal). Solo permiso de lectura "r - - " sería un 1 y dos ceros (100) que equivale a 4 en decimal (y en octal).

Por lo tanto, con tres cifras decimales podemos representar del 0 a las 7 combinaciones de permisos distintas.

Para modificar los permisos mediante chmod usando la especificación octal habría que escribir el siguiente comando:

```
sudo chmod 770 /permisos
```

Este comando otorga permisos completos (111 = 7) al usuario y grupo propietario mientras que mantiene al resto de los usuarios sin ningún permiso.

Simbólico

También podemos llamar a chmod con una indicación simbólica de los permisos. En este caso tendremos que distinguir 3 aspectos:

- Las letras u,g, o, a indican si estamos modificando los permisos de los usuarios, grupos, otros o todos respectivamente.
- Las letras r, w, x indican los permisos de lectura, escritura y ejecución respectivamente
- Los símbolos +, -, = indican si estamos otrogando, eliminando o indicando explícitamente un permiso.

De esta manera si escribimos el siguiente comando estaremos añadiendo el permiso de escitura al usuario propietario sobre la carpeta "/permisos"

```
sudo chmod u+x /permisos
```

También es posible otorgar varios permisos a la vez y a varios tipos simultáneamente.

```
sudo chmod ug+rw /permisos
```

En el comando anterior estamos asignando los permisos de lectura y escritura al usuario y grupo propietario de /permisos.

En otras ocasiones lo que queremos no es añadir ni quitar permisos sino establecerlo explícitamente. En este caso usamos el símbolo "=".

```
Sudo chmod a=rw /permisos
```

El commando anterior aplica permisos de lectura y escritura a todos los usuarios del sistema (el usuario, el grupo y los otros)

Consultar ACLs (CLI)

Desde la interfaz de comandos podemos consultar los permisos con un sencillo "ls -l" pero desde su salida solo veremos los permisos tradicionales. En caso de que el recurso tenga ACLs veremos un símbolo "+" al final de los permisos.

En esos casos es necesario la utilización del siguiente comando:

```
Getfacl /permisos
```

Este comando consulta los permisos y ACLs de la ruta indicada y nos devuelve información sobre el usuario propietario, el grupo principal y, como mínimo los permisos tradicionales en forma de ACLs.

Las ACLs tienen una estructura fija con 3 campos separados por dos puntos ":". El primer campo indica si la ACL se aplica sobre un usuario, grupo u otros. El segundo campo, si lo hubiera, indica el nombre concreto del usuario o grupo sobre el que actúa la ACL. Por último, el tercer campo contiene la combinación de permisos que se aplican en esa ACL.

Modificar ACLs (CLI)

Ya hemos visto como modificar permisos tradicionales usando el comando chmod. En caso de que tengamos ACLs aplicadas, el comando chmod no será suficiente y tendremos que usar el comando "setfacl". Modificar permisos requiere permisos administrativos por lo que las llamadas a este comando deben ir precedidas de "sudo" o bien hacerlas desde un usuario administrador o propietario del recurso.

Para añadir nuevas ACLs o modificarlas debemos escribir el siguiente comando:

```
sudo setfacl –m u:javi:rw– /permisos
```

Este comando añade una nueva ACL dándole permisos de escritura y lectura al usuario "javi" en la carpeta "permisos". Fijémonos en que estamos utilizando el parámetro "-m" para modificar o añadir ACLs

También es posible retirar una ACL usando el modificado "-x" y aciendo una referencia al usuario o grupo que queremos eliminar. En este caso no es necesario especificar los permisos ya que la eliminación de la ACL es completa.

```
sudo setfacl –x u:javi /permisos
```

Este comando borra la ACL que se aplica al usuario "javi" sobre la carpeta "/permisos"

Por último, también es posible eliminar todas las ACLs de un determinado recurso y dejarlo solamente con los permisos tradicionales de usuario y grupo propietario y otros.

```
sudo setfacl –b /permisos
```

Este comando elimina todas las ACLs de la carpeta "/permisos"

Orden de aplicación de las ACLs

Al utilizar ACLs es posible que generemos conflictos entre las propias ACLs. Hay que tener en cuenta que el usuario propietario es especial y que siempre debería tener el control total (recordemos que la propiedad se puede asignar a otro usuario con chown).

Para evitar los conflictos, los sistemas Linux aplican el siguiente orden en la aplicación de ACLs.

1. Permisos del propietario del recurso. Siempre existirá una regla para este usuario por lo que no le serán aplicables las ACLs siguientes.
2. Entrada específica para un usuario.
3. Permiso del grupo principal del archivo
4. Otros

Hay que tener en cuenta que la máscara puede limitar los permisos efectivos de todos los usuarios (a excepción del propietario)