



Nombre: .....

Fecha: / 11 /2010

Grupo: 1  2  3  4 

## PRÁCTICA 14

### GESTIÓN DE USUARIOS EN LINUX Y WINDOWS

Una de las grandes ventajas que históricamente han presentado los sistemas operativos basados en Unix (como GNU/Linux) con respecto a Windows ha sido la gestión de usuarios en las máquinas. Desde sus inicios Unix fue pensado como un sistema multiusuario, de tal modo que en cada máquina se podían "logar" diferentes usuarios con diferentes privilegios. Las últimas versiones de Windows han conseguido hacer la gestión de usuarios más sencilla y parecida a la de Linux (por ejemplo, Windows ofrece en sus últimas versiones la posibilidad de ejecutar aplicaciones en modo "Administrador", de modo similar a como hace Linux con el usuario "root"). También Windows reúne ahora a los distintos usuarios de un sistema operativo en grupos que comparten privilegios (tal y como se hace en Linux).

A lo largo de la práctica veremos cómo se pueden usar las herramientas (en general a través del intérprete de mandatos) para crear y eliminar nuevos usuarios y grupos, cómo comprobar los usuarios que hay en una máquina, cómo cambiar de usuario....

También prestaremos atención al usuario "root" o "Administrador". Tanto en Linux como en Windows, existe un usuario particular que tiene derechos especiales para crear nuevos usuarios, eliminarlos, arrancar ciertos servicios, instalar o desinstalar aplicaciones.... Sus capacidades son similares a la que definimos como "modo privilegiado" de acceso a la computadora (por tanto, debemos ser especialmente cuidadosos cuando hagamos uso de los mismos; por ejemplo, muchas de las vulnerabilidades de Windows se ponen de manifiesto cuando usamos la cuenta "Administrador", mientras que usando otras cuentas no pueden afectar a nuestro ordenador). El resto de usuarios, por lo general, sólo contarán con el "modo protegido" de acceso al ordenador.

A lo largo de la práctica también repasaremos algunos de los mandatos que hemos visto en prácticas anteriores; también usaremos algunos de los visores y editores de ficheros de texto más comunes en Linux y Windows.

1. Vamos a entrar en Linux con nuestra cuenta habitual (user = "alumno" y password = "alumno"). Abre una terminal. El usuario con el que te has logado en la máquina lo puedes saber fácilmente (está escrito en la parte izquierda del mensaje del "prompt"). ¿Qué más usuarios existen en la máquina? Vamos a visualizar el fichero "/etc/passwd" con un visor (no un editor) de ficheros.

Teclea en el intérprete de mandatos:

```
$man less
```

¿Para qué sirve el mandato "man"? (Puedes ejecutar "man man" en el intérprete para comprobarlo)

¿Para qué sirve el mandato "less"? (Lee su "Descripción" y apunta su finalidad. Quizá tengas que ejecutar también "man more" para aclararte).

2. Ahora que sabemos lo que hace "less", vamos a ejecutarlo sobre el fichero "/etc/passwd". Ejecuta en el intérprete:

```
$less /etc/passwd
```

En pantalla debe aparecer la lista de todos los usuarios que están dados de alta en el ordenador. La mayor parte de ellos se utilizan para tareas de administración o de gestión o comunicación interna de la máquina. Fíjate en el primero y el último que aparecen en la lista. Apunta todas sus características. ¿Qué significa cada una de ellas? (<http://blogubuntu.com/chuleta-para-etc-passwd/>).

3. Otro programa que se utiliza con asiduidad en el intérprete de mandatos es "ls". Ejecuta en el intérprete el mandato:

```
$man ls
```

A partir de su descripción, anota en tu informe para qué sirve el mismo. Una de sus opciones más usadas es "-l". Apunta también su utilidad.

4. Ejecuta el mandato:

```
$ls -l /etc/passwd
```

En la respuesta obtenida podrás observar información acerca del fichero "/etc/passwd". Ayudándote de lo explicado en <http://freeengineer.org/learnUNIXin10minutes.html#Listing>, apunta en tu informe el valor y el significado de cada uno de los atributos del fichero "/etc/passwd".

5. Vamos a intentar editar el fichero "/etc/passwd" por medio de un editor de texto de línea. Un conocido editor de textos en Unix y Linux es "nano". Ejecuta el mandato "man nano". Apunta alguna de las características señaladas en su descripción.

Ejecuta el mandato:

```
$nano /etc/passwd
```

El uso de "nano" es casi exclusivo a través de teclado (sin ratón). Las opciones que señala el editor como, por ejemplo, "^G", se refieren a la combinación de teclas "Ctrl + G". Moviéndote con las flechas del teclado trata de editar el contenido del fichero (por ejemplo, elimina un usuario por medio de la tecla "Supr"). Trata de guardar el fichero resultante ("^O") con el mismo nombre. Comprueba el resultado obtenido. Sal sin guardar el fichero.

¿A través de qué usuario estabas intentando editar el fichero? ¿A quién pertenece el fichero? ¿Qué permisos tenía asignados el fichero?

6. Otro de los ficheros relevantes a la hora de gestionar grupos y usuarios en Linux es el fichero "/etc/group".

Visualiza el fichero. Puedes utilizar el mandato "less". También puedes probar el mandato "cat"; si no sabes qué hace "cat", prueba primero "man cat". No deberías usar "nano", ya que nuestro propósito no es editar el fichero, sino sólo verlo.

Observa la primera y última líneas del fichero. Comprueba y anota con la ayuda de <http://www.cyberciti.biz/faq/understanding-etcgroup-file/> el significado de cada uno de sus campos.

Responde a las siguientes preguntas. ¿A qué grupos pertenece el usuario con el que te has logado en la máquina? ¿Puede haber más de un usuario en un grupo?

7. Veamos ahora cómo podemos agregar un nuevo usuario a nuestra máquina. Existen (al menos) dos mandatos que podrían hacer el trabajo, "adduser" y

"useradd". Comprueba el funcionamiento de los dos con "man". El primero de ellos es más fácil de usar. Teclea el mandato:

```
$adduser cuasi
```

¿Qué resultado obtienes? El usuario con el que estás logado no tiene permisos para dar de alta usuarios. Hace falta ejecutar el mandato como administrador. En algunas distribuciones Linux, al realizar la instalación, se crea un usuario "root" (o administrador) y su contraseña, el cual nos permitiría hacer esta tarea. En el caso de Ubuntu la gestión del usuario administrador ("root") es un poco distinta. Por defecto, al instalar el sistema operativo, no se puede acceder a la cuenta de "root" (ya que no se le ha asignado una contraseña). Sin embargo, el primer usuario que se ha creado al instalar Ubuntu pertenece al grupo "admin", y eso quiere decir que el mismo puede realizar tareas propias del usuario "root" para la administración del ordenador por medio de la orden "sudo".

Puedes encontrar más información sobre el mandato "sudo" en <http://es.wikipedia.org/wiki/Sudo>. Vamos a repetir el mandato anterior anteponiendo el mandato "sudo".

```
$sudo adduser cuasi
```

Completa los parámetros necesarios (presta atención a las contraseñas).

8. Vamos a volcar ahora los dos ficheros "/etc/passwd" y "/etc/group" a un tercer fichero. El mandato "cat" que hemos mencionado antes se puede usar para concatenar ficheros de texto. Comprueba el resultado de la siguiente orden:

```
$cat /etc/passwd /etc/group
```

Por defecto, la salida del mandato es la propia "shell" (o "bash"). Podemos modificar la salida por defecto, y hacer que la misma sea un fichero. Para ello tienes dos opciones:

```
$cat /etc/passwd /etc/group > usuarios_y_grupos_linux
```

El anterior mandato crea el fichero "usuarios\_y\_grupos\_linux" y vuelca el resultado de la operación "cat /etc/passwd /etc/group" en el mismo. Si el fichero ya existía, lo sobrescribe (perdiendo la anterior información).

```
$cat /etc/passwd /etc/group >> usuarios_y_grupos_linux
```

El anterior mandato crea el fichero "usuarios\_y\_grupos\_linux" y vuelca el resultado de la operación "cat /etc/passwd /etc/group" en el mismo. Si el fichero ya existía, añade el resultado al final del mismo.

Utiliza el mandato que prefieras de los dos anteriores. Consévalo para adjuntarlo a tu informe de la práctica.

¿A qué grupos pertenece el usuario que acabas de crear? ¿Pertenece al grupo "admin"?

9. Veamos hora cómo cambiar de usuario dentro de la misma "shell". Ejecuta el mandato "man su". Como puedes comprobar, el propio manual te informa de que está obsoleto y podría ser erróneo. Usa en su lugar el mandato "info su". ¿Para qué sirve el mandato "su"? Ejecuta el mandato:

```
$su cuasi
```

¿Qué ha sucedido en la parte izquierda del "prompt"? ¿En qué directorio te encuentras ahora?

10. Recuerda que el usuario "cuasi" no pertenece al grupo "admin". Veamos algunas de las consecuencias de lo mismo. Ejecuta el mandato:

```
$adduser cuasi2
```

¿Cuál es el resultado obtenido? Vuelve a intentarlo con el mandato que usamos antes:

```
$sudo adduser cuasi2
```

¿Qué ha sucedido ahora? ¿Quiénes son los "sudoers" (busca la respuesta en <http://es.wikipedia.org/wiki/Sudo>)?

No cualquier usuario de tu máquina puede hacer tareas de administración, sólo aquellos que pertenecen al grupo de los "sudoers" (por defecto, los miembros del grupo "root" y "admin").

11. Veamos ahora cómo podemos modificar los grupos a los que pertenece un usuario. Para ello de nuevo debemos ser administradores de la máquina, así que vuelve a "logarte" como "alumno" (\$su alumno). Observa que debería haber cambiado el prompt.

12. Vamos ahora a usar el mandato "usermod" para hacer que el usuario "cuasi" tenga privilegios de administrador (también podrías hacer uso del mandato "gpasswd"). Ejecuta "man usermod". Ejecuta ahora el mandato:

```
$sudo usermod -a -G admin cuasi
```

Comprueba en el fichero "/etc/group" (por medio de less, cat...) que la modificación ha tenido lugar. ¿Podrías haber editado "a mano" el fichero "/etc/group" para haber hecho la misma operación como administrador? ¿Y el fichero "/etc/passwd" para añadir un nuevo usuario?

Una de las características de Linux es que la mayor parte de la información se encuentra repartida en ficheros de texto en el sistema. Esto facilita que muchas operaciones, si sabemos dónde son almacenadas, se puedan hacer a través de mandatos, o también con un simple editor de texto.

13. Vuelve a usar "su cuasi". ¿Qué mensaje te muestra la "shell"? Repite el mandato que antes no se nos permitía hacer:

```
$sudo adduser cuasi2
```

¿Se te permite ahora completarlo?

Vuelve a redirigir el resultado de la operación "cat /etc/passwd /etc/group" al fichero "usuarios\_y\_grupos\_linux" y guarda el fichero (si no dispones de permisos de escritura sobre el fichero, utiliza alguno de los mandatos ya vistos como "sudo", "su"... para completar la tarea).

Vuelca también el resultado del mandato "history" a un fichero "mandatos\_practica\_14" y guarda el fichero.

14. Cierra la sesión gráfica y comprueba que puedes abrir una nueva sesión con cualquiera de los usuarios:

alumno  
cuasi  
cuasi2

15. Por último, antes de cerrar la sesión Linux, vamos a comprobar cómo un administrador puede cambiar la contraseña de cualquier usuario (incluida la del root). Para ello solo debe hacer (donde nombre\_de\_usuario debe ser el usuario al cual queremos modificar la contraseña):

```
$sudo passwd nombre_de_usuario
```

Compruébalo cambiando la contraseña del usuario "cuasi" (que también pertenecía al grupo "admin").

Vamos a intentar ahora realizar algunas operaciones similares a las que hemos completado en Linux dentro de Windows.

16. En primer lugar vamos a comprobar que en Windows también tenemos algunas de las facilidades que hemos usado en Linux. Si bien por defecto no disponemos de una versión de "less" instalada para Windows, sí que podemos disponer de una versión del más sencillo "more". Abre en tu máquina Windows un intérprete "cmd" y ejecuta el mandato:

```
>help more
```

¿Qué hace el mismo?

17. Busca un fichero de texto (\*.txt, \*.html, \*.xml, \*.eml...) en tu sistema y visualízalo por medio de "more".

18. Windows también dispone de editores de línea. Ejecuta el mandato:

```
>edit /?
```

¿Qué hace el programa "edit"?

19. Recupera el fichero que has usado en el ejercicio 17. Ejecuta el mandato:

```
>edit nombre_de_fichero
```

Observa que puedes editar y guardar el fichero de nuevo.

20. Windows también dispone de mandatos que nos permiten conocer los usuarios y los grupos que hay en nuestro ordenador. Ejecuta el mandato:

```
>net help  
>net help user  
>net help user | more
```

21. A continuación, ejecuta el mandato:

```
>net user
```

Comprueba que puedes redirigir el resultado del anterior mandato a un fichero, de modo similar a como lo hacías en el "shell" de Linux:

```
>net user > usuarios_y_grupos_windows
```

Comprueba el contenido del fichero por medio del mandato:

```
>more usuarios_y_grupos_windows
```

22. También podemos conocer los grupos de usuarios que están disponibles en nuestra máquina. Ejecuta el mandato:

```
>net localgroup
```

Repite el comando y redirige la salida al fichero "usuarios\_y\_grupos\_windows" (pero sin borrar el contenido anterior del fichero, recuerda cómo lo hicimos en la "shell" de Linux y decide si debes usar ">" ó ">>" para la redirección).

23. Al igual que nos pasaba en Linux, hay ciertas operaciones que son privilegio exclusivo del administrador. Comprueba el resultado de la siguiente operación:

```
>net user cuasi /add
```

El sistema debería denegarte el acceso a dicha operación. Vuelve a ejecutar una terminal "cmd" como administrador y repite la operación. Anota el resultado y explica la diferencia en tu informe. Asigna una contraseña al usuario:

```
>net user cuasi *
```

24. ¿A qué grupo ha sido añadido el nuevo usuario? Compruébalo por medio del mandato:

```
>net localgroup "nombre_de_grupo"
```

(Prueba con los que consideres más probables).

25. ¿Qué usuarios aparecen en el grupo de administradores?

```
>net localgroup Administradores
```

Ese es el motivo por el cual puedes ejecutar "cmd" en modo Administrador.

26. Vamos a cerrar la sesión, abrir sesión con un nuevo usuario y comprobar si los usuarios de tipo "Usuarios" también pueden ejecutar aplicaciones en modo administrador. Cambia de usuario y abre una sesión con el usuario "cuasi".

Trata de ejecutar una terminal en modo "Administrador". ¿Qué diálogo de usuario obtienes? Compáralo con el que obtenías en Linux al hacer una operación similar. Lo anterior debería haberte mostrado la importancia de tener todas tus cuentas, pero sobre todo aquellas que pertenezcan al grupo "Administrador", con una contraseña.

27. Cierra la sesión del usuario "cuasi" y vuelve a la sesión del usuario por defecto de la máquina. Vamos a ver cómo podemos hacer que el usuario "cuasi" pertenezca al grupo de los administradores.

Abre una terminal en modo administrador. Realiza la siguiente operación:

```
>net localgroup Administradores "cuasi" /add
```

Comprueba que la operación se ha completado con éxito:

```
>net localgroup Administradores
```

(Ahora debería aparecer "cuasi" en el grupo; desde el usuario "cuasi" podrías acceder a la cuenta "Administrador" sin ningún problema). Responde a la siguiente pregunta. ¿El hecho de que un usuario pertenezca al grupo de administradores quiere decir que ya tiene el control total sobre el sistema, o sólo quiere decir que puede acceder al ordenador, en determinadas circunstancias, como Administrador y realizar ciertas modificaciones? Piensa en lo que hemos ido haciendo en la práctica. Compáralo también con el uso de "root" y "sudo" en Linux. ¿La situación se parece más a tener un usuario "root" con el que podemos hacer cualquier cosa, o a tener un programa "sudo" que permite realizar ciertas operaciones de modo privilegiado?

28. Para terminar, vamos a eliminar al usuario "cuasi" de la lista de Administradores. Para ello puedes hacer uso del mandato:

```
>net localgroup Administradores "cuasi" /del
```

Comprueba que la operación se ha completado con éxito por medio de:

```
>net localgroup Administradores
```

29. Sube el informe de la práctica a tu página de inicio en belenus, junto con los ficheros usuarios\_y\_grupos\_windows, usuarios\_y\_grupos\_linux y mandatos\_practica\_14.