



Nombre:

Fecha: / 11 /2010

Grupo: 1 ☐ 2 ☐ 3 ☐ 4 ☐

PRÁCTICA 12

TRABAJANDO CON DISTINTOS SO. SHELL E INTÉRPRETE CMD.

En esta práctica empezaremos a trabajar con dos herramientas propias de Linux y Windows que nos permitirán gestionar llamadas a los Sistemas Operativos correspondientes sin necesidad de usar la interfaz gráfica de cada uno de ellos.

Estas herramientas son conocidas como "shell" o terminal en el entorno Linux y como intérprete de mandatos (ó "cmd") en Windows. Ambas aplicaciones entran en la categoría de lo que se llama "Interfaces de línea de mandatos" (CLI, Command Line Interface). Su importancia desde el punto de vista histórico es crucial, ya que durante bastante tiempo los SO (Linux, Unix, así como las versiones originales de MSDOS) no contaron con una interfaz gráfica, y era obligatorio gestionar todas las llamadas al SO a través de ellas. Hoy en día tanto Linux como Windows han evolucionado y permiten gestionar muchas de las funciones del sistema a través de las interfaces de usuario gráficas (GUI), pero aún así el uso de la shell en Linux sigue siendo muy extendido (no hay más que tener en cuenta que si arrancamos el ordenador en Linux en modo a prueba de fallos nos encontraremos con una shell; que el ordenador, al arrancar, primero ejecuta un CLI, y sobre éste ejecuta el entorno gráfico; o que muchas de las funciones que se hacen a través del entorno gráfico en realidad, de manera interna, son ejecutadas a través de la shell). Windows ha evolucionado de forma distinta, y las últimas versiones del mismo mantienen el intérprete de mandatos para poder realizar algunas funciones básicas de gestión de ficheros, particiones...pero muchas de las funciones del entorno gráfico no encuentran su equivalente en "cmd.exe" y por motivos de rendimiento no se ejecutan a través de mandatos de "cmd.exe".

No sólo por motivos históricos es conveniente conocer el uso de los CLIs. Sirva como ejemplo de su importancia que Windows, en las versiones posteriores a Vista, incluye (aunque no viene instalado por defecto) un programa llamado "Windows PowerShell" que permite interactuar con el SO, pero además también permite interactuar con programas de Microsoft como "SQL Server", "Internet Information Server" o la plataforma ".NET" (todo ello a través de una interfaz de línea de mandatos, <http://technet.microsoft.com/es-es/scriptcenter/powershell.aspx>). También es muy común el uso de CLIs como la shell de Linux para la configuración de servidores o máquinas que no van a disponer de una interfaz gráfica. Desde un punto de vista más amplio, también se utilizan CLIs para algunos programas que por diversos motivos no cuentan con una interfaz gráfica lo suficientemente potente (por ejemplo SAGE, MATLAB u otros muchos).

1. Ejecuta la máquina virtual de Ubuntu sobre VirtualBox. Pulsa la tecla "Esc" antes de que arranque la interfaz gráfica (mientras aparece en la pantalla el mensaje "GRUB loading"; puedes volver a reiniciar el equipo e intentarlo de nuevo si no lo has conseguido en la primera ocasión). De las tres opciones de arranque que te ofrece el menú GRUB, la primera (Ubuntu 8.04.1 kernel ... -generic) corresponde al arranque común del SO Ubuntu

8.04 (incluyendo su interfaz gráfica). La tercera sirve para comprobar la memoria de tu ordenador (en este caso de la máquina virtual). La segunda supone entrar en el modo a prueba de fallos. Elegimos la segunda. Apunta las tres opciones (y cómo has hecho para llegar hasta este menú) en tu informe de prácticas.

2. Ahora nos encontramos con un segundo menú. De las opciones disponibles, "resume" supone continuar con el arranque convencional (incluyendo el entorno gráfico de Ubuntu 8.04), mientras que la segunda y la cuarta tienen que ver con tareas administrativas.

La tercera opción nos va a permitir entrar en el shell del usuario root. Apunta las cuatro opciones disponibles y su significado en tu informe de prácticas.

Lee en el contenido de la siguiente referencia (<http://doc.ubuntu-es.org/Root>) el significado y misión del usuario root y apunta en tu informe de prácticas sus principales funciones (y las precauciones que hay que tomar para su uso). Escoge la opción "root" del menú y pulsa "ok".

3. Ahora te encontrarás con el "prompt" de la shell de Linux. El "prompt" te informa de que estás trabajando a través del usuario "root" en la máquina "alumno-desktop" (así que debes tener cuidado con los mandatos que ejecutes).

4. Ejecuta el mandato

```
#cd /home/alumno
```

(Trata de hacer uso tanto en la shell de Linux como en "cmd" en Windows de la tecla Tabulador, hará funciones de autocompleción de mandatos y de rutas de ficheros que te ahorrarán mucho tiempo).

5. Haz un listado del contenido de la carpeta en la que te encuentras:

```
#ls
```

Apunta el resultado en tu informe de prácticas. Si no conoces el funcionamiento o las opciones de un mandato de la consola, puedes tratar de recurrir al mandato "man nombre_de_mandato". "man" se encarga de mostrar las páginas de manual de ese mandato.

6. En la situación que te encuentras ahora (logado en la máquina como usuario "root") podrías llevar a cabo cualquier tarea sobre la misma (crear y eliminar carpetas, añadir y borrar usuarios, redefinir la contraseña del usuario "root"...). Crea una carpeta en el directorio en el que te encuentras de nombre "Datos_recuperados".

```
#mkdir Datos_recuperados
```

7. Comprueba (por medio del mandato "ls -l" que la carpeta ha sido construida). "ls" es el mandato para listar los contenidos de un directorio, la opción "-l" le indica que los muestre como si fueran una lista.

Apunta en tu informe la siguiente información. ¿Qué permisos tiene la carpeta "Datos_recuperados"? (Se corresponden con la cadena "r w x") ¿Qué nombre de usuario aparece asociado a la carpeta "Datos_recuperados"? ¿Qué nombre de grupo? ¿Qué nombres de usuario y de grupo aparecen asociados a las carpetas restantes?

8. Vamos a crear ahora un nuevo usuario en nuestra máquina con los datos de nuestra CUASI (si no los recuerdas puedes usar otro usuario y contraseña distintos).

```
#adduser Usuario_CUASI
```

¿Qué acciones han tenido lugar? Apúntalas en tu informe. Asígnale una contraseña (la de tu CUASI u otra distinta, pero no la olvides).

Ante el mensaje de la "shell" "Enter new UNIX password:" teclea una contraseña para ese usuario (no hace falta que sea la de tu CUASI, pero recuérdala para el resto de la práctica). Completa los datos restantes para la creación del usuario.

(Si te equivocas al crear el usuario, puedes eliminarlo por medio del uso de `#userdel nombre_de_usuario`).

9. Veamos ahora cómo podemos recuperar los mandatos que han sido usados en nuestra sesión. Ejecuta el mandato:

```
#history
```

En pantalla deberían aparecer todos los mandatos que has ejecutado durante la sesión. Esa misma información, que por defecto aparece en el propio shell, puede ser "volcada" o redirigida a otro medio distinto. Dirígete a la carpeta `"/home/Usuario_CUASI"` (por medio del mandato `"cd /home/Usuario_CUASI"`). Ejecuta ahora el mandato:

```
#history >> mandatos_primera_parte
```

Vamos a reiniciar el equipo por medio del mandato:

```
#reboot
```

10. Cuando aparece la pantalla de inicio de Ubuntu, puedes logarte por medio de dos usuarios. Por una parte, el usuario por defecto del que disponen las máquinas de la sala es:

Usuario: alumno
Contraseña: alumno

También está disponible el usuario:

Usuario: Usuario_CUASI
Contraseña: la que has utilizado al crear el usuario

11. Accede a la máquina por medio del segundo de los usuarios. Vamos a comprobar que las carpetas y ficheros que hemos creado desde la shell también están disponibles desde la GUI. Dirígete a la pestaña "Lugares" y dentro de la misma a "Carpeta personal". ¿Aparece un fichero llamado "primera_parte"? Dirígete también a la carpeta "/home/alumno/". Comprueba si ha aparecido una nueva carpeta "Datos_recuperados".

12. En realidad, en Ubuntu, aparte del entorno visual, uno puede trabajar fácilmente desde fuera del mismo. Existen diversas terminales de texto a disposición del usuario. Utiliza el comando "ctrl + alt + F1" (o F2,..., F6). Puedes logarte en cualquiera de ellas y utilizarlas de forma simultánea.

Puedes volver a la interfaz gráfica por medio de "ctrl + alt + F7".

13. Vamos a realizar ahora gestión de usuarios en Windows desde el intérprete de mandatos. Abre un intérprete cmd en Windows (lo tienes que abrir en modo "Administrador"). Utiliza el siguiente mandato:

```
>net user
```

Apunta en tu informe la lista de usuarios que aparecen en la máquina Windows.

14. Existen mandatos también para crear y eliminar usuarios. Prueba el siguiente mandato:

```
>net user Usuario_CUASI /add
```

Apunta el mandato correspondiente en tu informe de la práctica. Comprueba por medio del mandato "net user" que el usuario ha sido creado. Veamos cómo le podemos asignar una contraseña al usuario:

```
>net user Usuario_CUASI *
```

Apunta el mandato en tu informe. Ahora debes escribir la contraseña para ese usuario dos veces.

15. Sin apagar el ordenador (recuerda que se perderían todos los cambios), elige la opción "Cambiar de Usuario" del menú "Apagar" de Windows ¿Qué usuarios aparecen en el menú? Accede con el nuevo usuario creado.

16. Vuelve a "Cambiar de usuario" y elige el usuario original de la máquina.

17. De la misma forma como has creado usuarios, puedes eliminarlos por medio del mandato (no hace falta que lo uses, es sólo para tu conocimiento):

```
>net user usuario /delete
```

Vamos a pasar ahora a ver cómo podemos usar variables de entorno en Linux y en Windows desde el intérprete de mandatos. Las variables de entorno son alias o sustitutos de ciertas propiedades básicas del sistema que se utilizan en programación o en administración de sistemas. Uno de

sus usos comunes es dar un nombre corto (por ejemplo TEMP) a recursos que el sistema utiliza con asiduidad ("C:\Users\Administrador\AppData\Local\Temp").

18. Ejecuta en una shell (la puedes encontrar en "Aplicaciones -> Accesorios -> Terminal") de Ubuntu el mandato:

```
$set
```

Es probable que su salida sea tan grande que no hayas podido ver gran parte de ella. Hay diversas formas de tratar de solucionar lo mismo.

En primer lugar, y por medio de la combinación de teclas "Mayúsculas + RePág" o "Mayúsculas + AvPág", puedes subir y bajar por la salida de la shell. Por medio de la combinación "Mayúsculas + RePág" trata de recuperar el principio de la salida del mandato "set" (debería ser la línea en la que escribiste "\$set").

Como podrás observar, ni siquiera así has conseguido llegar al origen de la respuesta del sistema. Hay otras formas de conseguir que la salida de un mandato se produzca de forma gradual.

Ejecuta ahora el siguiente mandato:

```
$set | less
```

Ahora puedes controlar la salida de código por medio de la tecla "Return" o por medio de "RePág". El programa "less" pertenece al proyecto GNU (<http://es.wikipedia.org/wiki/GNU>), y es un visor para ficheros. Lo que hemos hecho en el anterior mandato es coger la salida del comando "set" y redirigirla al programa "less", que nos la ha mostrado de forma graduada (en cualquier momento puedes "salir" de la aplicación "less" presionado la tecla "q", abreviatura para "quit").

También hemos visto una tercera forma de "controlar" la salida de un mandato. Ejecuta el siguiente mandato:

```
$set >> variables_de_entorno
```

¿Qué hemos hecho con el anterior mandato?

Ejecuta ahora:

```
$gedit variables_de_entorno
```

¿Qué programa se ha ejecutado? ¿Pertenece al entorno gráfico o a las aplicaciones del modo de línea?

Anota en tu informe de prácticas el valor de las variables "SHELL", "HOME", "USER", "PATH", "PS1", "HOSTNAME" y "LANG". Apunta también en tu informe qué significa cada una de ellas.

19. Existe otra forma de conocer el valor de una variable de entorno que no requiere comprobar todo el contenido del mandato "set".

Ejecuta en la terminal:

```
$echo $PATH  
$echo $HOME  
$echo $USER  
$echo $HOSTNAME
```

Compara los valores obtenidos con los del ejercicio anterior (el carácter "\$" es un carácter especial de la shell que indica que el parámetro dado, por ejemplo PATH, debe ser sustituido por su valor).

20. Veamos ahora cómo podemos ejecutar otros programas desde el intérprete de mandatos. Escribe el siguiente mandato en la shell:

```
$openoffice variables_de_entorno
```

¿Qué ha sucedido? ¿Qué programa se ha ejecutado? Comprueba el texto y cierra el programa.

21. Para terminar con la parte de Linux, ejecuta en la terminal el mandato

```
$history >> mandatos_segunda_parte
```

Asegúrate de que conservas los ficheros de texto "mandatos_primera_parte" y "mandatos_segunda_parte", ya que debes colgarlos junto con el informe de la práctica 12 (puedes sacarlos de la máquina virtual por mail, montando una unidad de Windows y pasando a ella los ficheros, por ftp a belenus directamente desde la máquina virtual...).

22. Pasamos ahora a trabajar con el "cmd" de Windows. Ejecuta el mandato:

```
>set
```

23. Anota en tu informe de prácticas el valor de las variables de entorno "HOMEPATH", "USERNAME", "USERDOMAIN", "SystemDrive".

24. También en el intérprete "cmd" podemos conocer el valor de variables de entorno por medio de "echo". Sin embargo, a la hora de referirnos a una variable de entorno, mientras que en Linux nos referíamos a ella con "\$" para que la misma fuese evaluada, en Windows deberemos usar "%". Escribe en el intérprete de mandatos:

```
>echo %HOMEPATH%  
>echo %USERNAME%  
>echo %USERDOMAIN%  
>echo %SystemDrive%
```

Compara sus valores con los antes obtenidos.

25. También en "cmd" podemos volcar la salida de un mandato a una salida que no sea la pantalla (por ejemplo, a un fichero). Ejecuta el siguiente mandato:

```
>set >> variables_de_entorno
```

26. Para comprobar el contenido del fichero con un editor, puedes hacer uso del mandato:

```
>notepad variables_de_entorno
```

27. Una de las variables de entorno que más se suele utilizar, y que conviene conocer, es la variable "Path". Tanto en Windows como en Linux, la variable PATH indica todos los directorios en los que el sistema busca los mandatos que se ejecutan a través del intérprete.

Dentro del intérprete ejecuta el mandato:

```
>winword
```

Deberías obtener un error. El mismo se debe a que el intérprete de mandatos no es capaz de encontrar el archivo ejecutable "winword.exe" (porque su carpeta no está añadida a la variable de entorno "PATH"). Añade el siguiente mandato en el intérprete:

```
>set PATH=%PATH%;C:\Program Files\Microsoft Office\OFFICE11
```

Comprueba el valor de PATH tras la modificación ("echo %PATH%").

28. Ejecuta ahora el mandato:

```
>winword variables_de_entorno
```

¿Qué ha sucedido? ¿Qué programa se ha ejecutado? ¿Por qué antes no funcionaba?

29. Sube y enlaza desde tu página "index" el informe de la práctica, así como los ficheros "mandatos_primera_parte" y "mandatos_segunda_parte".