

Practica 3.1

En esta primera práctica los alumnos realizarán ejercicios de consolidación y refuerzo práctico del aprendizaje en relación con los contenidos del tema 2 sobre el soporte de sistema operativo UNIX. En particular, sobre el uso básico del sistema a través de su interfaz de comandos y el sistema de archivos.

Los ejercicios que se proponen pueden realizarse en cualquier distribución o versión de sistema operativo UNIX, preferiblemente con bash como Shell de usuario. Las respuestas y resultados de los ejercicios 1 y 2 se editarán en un fichero de texto simple (NO con un procesador de texto como Word). Preferiblemente se utilizarán los editores de texto Vi o Sublime Text 3. El fichero se entregará a través de Moodle como respuesta a la actividad configurada en Moodle por los profesores de la asignatura. El trabajo se realizará preferentemente por grupos de 2 alumnos.

El fichero de respuestas incluirá una cabecera con un título y el nombre completo y DNI de los autores del informe. La entrega en Moodle la realizará un solo miembro de la pareja.

Se valorará la corrección conceptual y la claridad y capacidad didáctica, junto con la buena redacción y estructura del documento.

Ejercicio 1

Haga login en el sistema y abra una ventana de comandos (Terminal) para ejecutar los comandos propuestos en las transparencias 52 y 53 de del tema de UNIX de la asignatura. El uso del comando "vi" en la transparencia 52 tiene por propósito editar el fichero core.txt para que su contenido sea el que se refleja en la transparencia 53.

1. Describa brevemente el propósito y resultado de cada uno de los comandos ejecutados
2. ¿Para qué cree que sirve el operador ">" utilizado con el comando "echo" ?
3. ¿Qué diferencia hay en el resultado de ejecutar estos tres comandos: "ls", "ls -l", "ls -li" ?
4. En relación con la salida del comando "od -cb", ¿Qué representa la secuencia "\n" ? ¿Cómo se representan en octal los caracteres "blanco" y "."?
5. El comando "touch" se utiliza para actualizar la fecha de último acceso y modificación del fichero cuyo nombre se le pase como argumento. Como utilidad adicional, si el fichero no existe lo crea vacío. Utilice el comando "touch" para crear un fichero de nombre ".invisible". Liste el contenido del directorio en que se encuentra con "ls" y "ls -a". ¿Qué diferencia hay?

Ejercicio 2

Ejecute los comandos de los ejemplos de la transparencia 54.

1. Describa brevemente el propósito y resultado de cada uno de los comandos ejecutados
2. Sitúese en el directorio *home* de su cuenta e indique el *path* relativo y el *path* absoluto del directorio de nombre "directorio" recién creado
3. El argumento 600 del comando "chmod" es un número octal, conviértalo a un número binario de 9 bits y ayúdese de esta representación para explicar el efecto del comando.
4. ¿Qué diferencia hay en el resultado de ejecutar los siguientes comandos: "ls -l directorio", "ls -ld directorio"?

Ejercicio 3

Este ejercicio no contiene preguntas a contestar en el fichero de respuestas de la práctica que se entrega vía Moodle.

En este ejercicio vamos a realizar la configuración inicial de la aplicación UNIX Git¹. Git es un sistema de control de versiones y gestión de repositorios de software que soporta el desarrollo colaborativo distribuido de sistemas software. Git se utilizará a lo largo de la asignatura para la entrega y gestión de configuración de las prácticas propuestas. En particular, se utilizará un portal web de *hosting* de repositorios Git llamado github (<https://github.com>).

¹ <http://git-scm.com/book/en/Getting-Started-Git-Basics>

Desde el punto de vista de UNIX, Git es un comando más, con una sintaxis y modo de interacción análogos al resto de comandos del sistema operativo.

1. Compruebe si Git está instalado en su máquina y, en su caso, en qué directorio mediante `"which git"`.
2. Si no está instalado, puede instalarse ejecutando (con privilegios de superusuario) el comando `"apt-get install git"` (Ubuntu).
3. Regístrese como usuario de github (<https://github.com/join>)
4. Configure en Git su nombre de usuario y dirección de correo electrónico mediante los siguientes comandos (pueden ser los mismos datos utilizados en el registro en github)
 - `git config --global user.name "Your Name Here"`
 - `git config --global user.email "your_email@youremail.com"`
5. Compruebe que se ha creado el fichero de configuración `"~/.gitconfig"`.
6. Las interacciones e intercambios de ficheros entre su repositorio Git local y el repositorio Git remoto en github se realizan a través de un canal cifrado. Esto se puede hacer mediante el protocolo ssh (Secure Shell), utilizado para ejecutar comandos en máquinas UNIX remotas. El algoritmo de cifrado de ssh se denomina RSA y se basa en la generación de una pareja de claves, una privada y secreta y la otra pública. Siga las siguientes instrucciones² para generar las claves RSA y el registro de su clave pública en github.
 - Compruebe si ya posee claves RSA en su cuenta haciendo `"cd ~/.ssh"` para comprobar si ese directorio ya existe. Si ya tiene claves creadas puede utilizarlas y registrar su clave pública en github como se indica a continuación, sin crear claves nuevas.
 - Genere las claves RSA asociadas a su dirección de correo electrónico mediante el comando: `"ssh-keygen -t rsa -C "your_email@youremail.com"`.
 - Compruebe el contenido del directorio `"~/.ssh"`, su clave pública está en el fichero de nombre `id_rsa.pub`
 - Copie su clave pública en el perfil de su cuenta en github (<https://github.com/settings/profile>), en el apartado SSH Keys. Para evitar errores por introducción de caracteres no imprimibles, como `"\n"`, puede copiar la clave pública al *clipboard* del entorno gráfico de LINUX mediante el comando: `"xclip -sel clip < ~/.ssh/id_rsa.pub"` (equivalente a hacer copy con el ratón)
 - Seleccione *Add SSH key* en su perfil de github y pegue (paste, botón derecho del ratón) su clave pública
 - Confirme, en su caso, con la contraseña utilizada al crear las claves con `ssh-keygen`
 - Confirme que puede conectarse al servidor de github mediante: `"ssh -T git@github.com"`.
 - Este comando requerirá usar la password que se utilizó para crear las claves y producirá una salida similar a la siguiente:

```
Hi jcyelmo! You've successfully authenticated, but GitHub does
not provide shell access.
```

- Puede leer un tutorial básico sobre Github en: <https://guides.github.com/activities/hello-world/>

² Instrucciones detalladas en: <https://help.github.com/articles/connecting-to-github-with-ssh/>