Tema

1.4

Sistema Operativos (SSOO)

Shell Scripts I

Universidad Francisco de Vitoria
UFV Madrid

Grado en Ingeniería Informática
Escuela Politécnica Superior

Índice

- Bash
- Shell scripts

Introducción a Bash

Es la interfaz entre el usuario final y el Sistema Operativo.

- **No es el S.O**.
- Existen múltiples versiones y podemos averiguar cual tenemos instalada haciendo.

Existen múltiples Shells en Unix:

- Bourne shell (**sh**), C shell (**csh**), Korn shell (**ksh**), TC Shell (**tcsh**)
- Bourne Again shell (**bash**).

La más popular es la "**bash**" shell.

- **bash** incorpora las prestaciones más útiles de la Korn shell (ksh) y la C shell (csh).
- Ofrece mejoras funcionales sobre otras shells desde el punto de vista de programación y de su uso interactivo.

Introducción a Bash

Consultar la versión de la Shell:

$echo $SHELL

$bash --versión

$echo $BASH_VERSION

$whereis bash

$cat /etc/shells

```
mcardenas@mcardenas:~$ echo $SHELL
/bin/bash
```

```
mcardenas@mcardenas:~$ echo $BASH_VERSION
4.4.20(1)-release
```

Cambiar la Shell actual a bash:

$chsh –s /bin/bash

Introducción a Shell Scripts

bash no es únicamente una excelente shell por línea de comandos.

- Es un lenguaje de scripting que permite utilizar las capacidades de la shell para automatizar muchas tareas que implicarían distintos comandos introducidos de manual.

Scripting no es un lenguaje de programación:

- Los lenguajes de programación son por lo general, más potentes y mucho más completos que los lenguajes de scripting.
- Los lenguajes de programación parten del código fuente, el cual luego es compilado para crear los ejecutables.
- Un ejecutable permite que los programas sean portables entre diferentes SSOO.

Comandos Linux

- chmod (Álvaro Ortiz) Enseña a usarlo
- rm -rf (Andrés Arcones)
- cd $HOME (Guillermo López)
- mv (Ernesto Marín)
- chown (Enrique Ruiz) Enseña a usarlo
- ls –al (Fernando Pérez)
- clear, echo (Eduardo Briales)
- tree, lstree (Julio Miranda) Enseña a instalarlo
- cat, nano (Silvia Romero) Enseña a usarlo
- who –u/b (Marta Meneses)
- whereis, pwd (Blanca Herreros)
- ps –a | grep bash (Marcos Rodríguez)

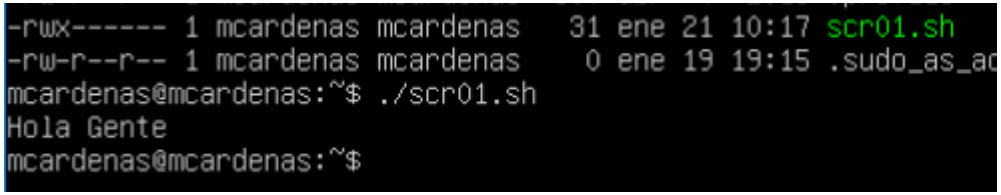
Introducción a Shell Script

- Un lenguaje de scripting también comienza por el código fuente, pero no se compila en un ejecutable
- En su lugar, un intérprete lee cada instrucción del fichero fuente y las ejecuta de forma secuencial.
- Los programas interpretados son por lo general son más lentos que los compilados.

Ejecuta tu primer script

Crea un fichero llamado “scr01.sh”:

```
#!/bin/bash
echo “Hola Mundo”
```



Interpretar: indicar que el fichero es interpretado por **bash** para su ejecución

Permisos: es importante hacer que el script sea ejecutable:

$ chmod 700 scr01.sh

$ ls –l: -rwx------ scr_01.sh

Introducción a Shell Script

Ejecuta tu segundo script

Crea un fichero llamado “scr02.sh”:

```
#!/bin/bash
mkdir log_apache
mkdir log_tomcat
cp ./servers/server1/*.log log_apache/
cp ./servers/server2/*.log log_tomcat/
```

$ chmod 700 scr02.sh

$ ls –l: -rwx------ scr_02.sh

```
mcardenas@mcardenas:~$ ./scr02.sh
mcardenas@mcardenas:~$ tree servers
servers
├── server1
│   ├── apache1.log
│   └── apache2.log
└── server2
    ├── tomcat1.log
    └── tomcat2.log

2 directories, 4 files
mcardenas@mcardenas:~$ tree log_apache
log_apache
├── apache1.log
└── apache2.log

0 directories, 2 files
mcardenas@mcardenas:~$ tree log_tomcat
log_tomcat
├── tomcat1.log
└── tomcat2.log

0 directories, 2 files
mcardenas@mcardenas:~$ _
```

Shell Script: variables

- Los valores se almacenan como cadenas de texto.
- No es necesario declarar las variables, con asignarle un valor es suficiente.
- Los operadores matemáticos que convierten las variables en número para realizar cálculos.
- Para consultar el valor de una variable se antepone el símbolo **$** al nombre.

```
#!/bin/bash
cadena = “¡Hola Gente!”  #esto es un comentario
echo $cadena
```

- No existe el casting de los tipos de las variables, por lo que se podrá asignar a una misma variables diferentes valores sin importar su tipo.
- Se recomienda usar el mismo tipo de datos a una variable dentro de un mismo script.
- El carácter de escape de la bash **** mantiene el carácter siguiente.

Shell Script: uso de comillas

- Las cadenas de textos deben ir acompañadas de comillas simples o dobles.
- El uso de comillas dobles "…" permite referenciar otras variables dentro de la cadena, e.g. cadena = "Tu edad es **$edad**"
- El uso de comillas simples permite mostrar una cadena sin la posibilidad de referenciar otras variables dentro de la misma cadena.
  - cadena = 'Tu edad es $edad'

Ejercicio:

Mostrar la siguiente cadena de texto:

El valor de la variable 'edad' es "31"

Shell Script: variables entre procesos

- Las cadenas de textos deben ir acompañadas de comillas simples o dobles.
- El uso de comillas dobles "…" permite referenciar otras variables dentro de la cadena, e.g. cadena = "Tu edad es **$edad**"
- El uso de comillas simples permite mostrar una cadena sin la posibilidad de referenciar otras variables dentro de la misma cadena.
  - cadena = 'Tu edad es $edad'

Bibliografía

- **CARRETERO**, Jesús, GARCÍA, Félix, DE MIGUEL, Pedro, PÉREZ, Fernando. Sistemas Operativos: una visión aplicada. McGraw-Hill, 2001.
- **STALLINGS**, William. Sistemas operativos: aspectos internos y principios de diseño. 5ª Edición. Editorial Pearson Educación. 2005. ISBN: 978-84-205-4462-5.
- **TANENBAUM**, Andrew S. Sistemas operativos modernos. 3ª Edición. Editorial Prentice Hall. 2009. ISBN: 978-607- 442-046-3.

marlon.cardenas@ufv.es

Universidad Francisco de Vitoria
UFV Madrid

Grado en Ingeniería Informática
Escuela Politécnica Superior