**Comandos de linux**

* BASH
  + *local MESSAGE="${@}"* : variable local (dentro de una función)
  + *readonly VERBOSE='true'* : solo lectura
  + *let NUM='2 + 3'* : NUM = 5
  + *expr 1 + 1* : 2
  + *NUM=$(expr 2 + 3)* : NUM = 5
  + *${0}* : nombre del último comando ejecutado
  + *${#}* : número de argumentos del comando
  + *${?}* : exit status del último comando
  + *${@}* : argumentos del comando ejecutado en forma de lista
  + *${\*}* : argumentos del comando ejecutado en forma de string
* SHELL
  + *!v* : comando más reciente que empieza por v
  + *logger 'Hello from the command line!'* : añade una traza al fichero messages
  + *sudo tail /var/log/messages* : últimas 10 líneas del fichero messages
  + *logger -t my-script 'Tagging on.'* : etiqueta la traza con el nombre my-script
  + *bc -l* : calculadora bash
  + *which -a head* : busca las rutas donde está el comando head
  + *type head* : donde se ejecuta el comando head
  + *type -a for* : nos dice que for es una palabra del shell
  + *help hash | less* : ayuda del comando del sistema hash
  + *man dirname* : manual de dirname
  + */usr/bin* : en esa carpeta están los ejecutables que se pueden ejecutar desde cualquier PATH a nivel usuario
  + */usr/sbin* : en esa carpeta están los ejecutables que se pueden ejecutar desde cualquier PATH a nivel administrador
  + *basename /vagrant/luser-demo06.sh* : devuelve el nombre sin el PATH
  + *dirname /vagrant/luser-demo06.sh* : devuelve el PATH sin el nombre del fichero
  + *sleep 1* : "dormir" durante un segundo
  + *tail add-local-user.sh -c 7* : extrae los últimos 7 caracteres del fichero indicado
  + *head add-local-user.sh -c 7* : extrae los primeros 7 caracteres del fichero indicado
  + *locate userdel* : localiza userdel para el usuario loggado
  + *touch userdel* : crear un archivo userdel
  + *sudo updatedb* : reindexa los archivos
  + *locate userdel | grep bin* : localiza userdel con el path que contenga bin para el usuario loggado
  + *sudo locate .bashrc* : localiza .bashrc para todos los usuarios
  + *find /usr/bin/\*log\** : busca los ficheros con patrón \*log\* en el path indicado
  + *find /usr/sbin/ -name userdel* : busca el fichero con nombre userdel en el path
  + *id usuario* : muestra el id del usuario
  + *sudo userdel -r jlocke* : borra el usuario incluido su directorio
  + *tar -cf catvideos.tar catvideos/* : comprimir en catvideos.tar el contenido de catvideos/
  + *tar -xf ../catvideos.tar* : extraer el contenido de catvideos.tar
  + *sudo chage -E 0 woz* : contraseña del usuario woz caduca ahora mismo
  + *cut -c 1-4 /etc/passwd* : extraer desde el carácter 1 al 4 del fichero etc/passwd
  + *echo -e 'one\ttwo\tthree' | cut -f 1* : extrae la primera palabra de una cadena separada por tabulaciones
  + *echo 'one,two,three' | cut -d ',' -f 3* : extrae la tercera palabra de una cadena separada por comas
  + *cut -d ':' -f 1,3 --output-delimiter=',' /etc/passwd* : extrae los campos 1 y 3 y sustituye el delimitador ‘:’ por ‘,’
  + *grep lastly people.csv* : localiza las líneas del fichero donde exista ‘lastly’
  + *grep -v '^first,last$' people.csv* : obtener todas las líneas del fichero excepto las que empiecen por ‘first’ y acaben por ‘last’
  + *awk -F ':' '{print $1, $3}' /etc/passwd* : es parecido a *cut -d ':' -f 1,3 /etc/passwd*
  + *awk -F ':' -v OFS=',' '{print $1, $3}' /etc/passwd* : obtener todas las líneas extrayendo los campos 1 y 3, con el delimitador ‘:’, y sustituyéndolo por ‘,’
  + *awk -F ':' '{print $1"@"$3}' /etc/passwd* : entre campo 1 y 2, escribe ‘@’
  + *awk '{print $1, $2}' lines* : extrae campo 1 y campo 2 del fichero lines, separado por espacio(s)
  + *echo -e 'hola\npepo'* : imprimir con caracteres especiales (salto de línea)
  + *ping -c 3 10.9.8.11* : prueba de conexión a la ip recibiendo 3 paquetes
  + *echo '10.9.8.12 server02' | sudo tee -a /etc/hosts* : escribe en el fichero hosts el contenido de echo por la STDIN
  + *$(cat servers)* : salida del comando cat servers
  + *pwd* : nombre del directorio de trabajo actual
* STDOUT, STDIN, STDERR
  + *id -un > id* : la salida del comando la redirige a un fichero que se llama "id" (STDOUT)
  + *read LINE < ${FILE}* : redirige a una variable LINE el contenido del fichero contenido en la variable FILE (STDIN)
  + *echo "${RANDOM} ${RANDOM}" >> ${FILE}* : redirige al fichero FILE, pero añadiendo, no lo machaca
  + *read LINE 0< ${FILE}* : redirige STDIN a una variable, usando FD 0.
  + *head -n3 /etc/passwd 1> ${FILE}* : redirige STDOUT a un fichero usando FD 1. machacando el fichero.
  + *head -n3 /etc/passwd /fakefile 2> ${ERR\_FILE}* : redirige STDERR a un fichero usando FD 2.
  + *head -n3 /etc/passwd /fakefile &> ${FILE}* : redirige STDOUT y STDERR a un fichero.
  + *head -n3 /etc/passwd /fakefile |& cat -n* : redirige STDOUT y STDERR a través de un pipe.
  + *echo "This is STDERR!" >&2* : enviar salida a STDERR
  + *head -n3 /etc/passwd /fakefile > /dev/null* : descartar STDOUT
  + *head -n3 /etc/passwd /fakefile 2> /dev/null* : descartar STDERR
  + *head -n3 /etc/passwd /fakefile &> /dev/null* : descartar STDOUT y STDERR