



 Grado en Ingeniería Informática

**Seguridad y Riesgos en Sistemas de Información**

Práctica 2 - Copias de Seguridad

Laboratorio

## Contenidos

<b>Introducción</b>	<b>3</b>
<b>Temporización</b>	<b>3</b>
<b>Entorno</b>	<b>3</b>
Herramientas . . . . .	3
Binarios . . . . .	4
<b>Funcionalidad</b>	<b>4</b>
Estructura de Archivos . . . . .	5
Consideraciones . . . . .	6
Programas de Utilidad . . . . .	6
bcklink . . . . .	6
bckview . . . . .	6
bckmv . . . . .	6
bckrestore . . . . .	7
bckrm . . . . .	7
Consideraciones . . . . .	7
<b>Entrega</b>	<b>7</b>

## Introducción

Tener copias de seguridad de los archivos del sistema es algo fundamental en cualquier entorno de trabajo. Con copia de seguridad nos referimos a una o varias copias de los artefactos seleccionados en un formato que permita su gestión y recuperación en caso de que el sistema llegue a un estado no deseado.

En cualquier sistema, y especialmente en los sistemas linux, de entre los archivos que componen el sistema cobran especialmente importancia los archivos de configuración. Estos archivos se utilizan para configurar tanto aspectos generales del sistema, como la gestión de los usuarios, configuración de las sesiones, o aspectos relacionados con aplicaciones en particular. Estos últimos, para un sistema *ubuntu*, suelen ubicarse en el directorio `~/.config`.

La ventaja de estos archivos de configuración es que, en su mayoría, son archivos en texto plano. Permiten su gestión de forma eficiente para, no sólo editar, sino también copiarlos o guardarlos para su posterior uso en el mismo u otros sistemas.

Una práctica que nos ayuda a mantener la consistencia y seguridad de un sistema es poder guardar o restaurar estos archivos de configuración bajo demanda. De esta forma podemos tener copias de estos para su posterior restauración o consulta en caso de que sea necesario, especialmente si se borran o se quiere restaurar el sistema después de una nueva instalación.

Esta práctica se realiza en parejas y tiene una valoración de 1 punto (sobre un total de 2,5 en las prácticas).

Al terminar la práctica se habrán trabajado los siguientes puntos:

- ☐ Manejo de comandos relacionados con la gestión de archivos
- ☐ Gestión de copias de seguridad de archivos
- ☐ Desarrollo de scripts en *bash script*

## Temporización

3 semanas (ver cronograma de la asignatura).

- Semana 1: planteamiento
- Semana 2: dudas
- Semana 3: entrega y evaluación

## Entorno

### Herramientas

Los comandos propuestos para la gestión del sistema de copia de seguridad son los siguientes:

- `basename`
- `cut`
- `echo`
- `grep`
- `ln`
- `mv`
- `printf`
- `readlink`

También se puede utilizar cualquier estructura de control.

## Binarios

La variable `$PATH` contiene las rutas donde buscar los comandos a ejecutar. Normalmente, en un entorno *ubuntu*, si el directorio `~/local/bin` existe este es añadido al path para el usuario correspondiente. A partir de ese momento, cualquier ejecutable que se encuentre en ese directorio puede ser ejecutado desde cualquier ubicación.

Para el desarrollo de la práctica los comandos de utilidad del sistema se crearán en esta carpeta y serán *scripts* en *bash*.

## Funcionalidad

El programa de copias de seguridad a implementar consiste en una colección de comandos de utilidad que gestionan la salvaguardia de los archivos de configuración, o cualquier archivo o directorio, que el usuario desee. Siendo más precisos, lo que realiza el sistema de copia de seguridad es copiar los archivos que gestiona en una carpeta centralizada y sustituir los ficheros originales por enlaces simbólicos a estos. De esta forma es trivial crear una copia, en la nube o con un sistema de control de versiones, de dicha carpeta.

El conjunto de la utilidad se denomina **bck**. No existe el comando como tal, pero nos servirá de nomenclatura para referirnos *al sistema* de copia y gestión de archivos en general.

El sistema, *bck* de ahora en adelante, consta de dos partes principales:

- Una estructura de archivos donde se organizan las copias de los archivos
- Un conjunto de programas de utilidad que gestionan la estructura de archivos

La estructura de directorios funciona a modo de *base de datos* que almacena los ficheros que forman parte de la copia. Cuando se añade un fichero, refiriéndonos a ficheros o directorios en general, a este se le asocia un *identificador*, *id* de ahora en adelante. Este *id* identifica al fichero de forma unívoca en el sistema y permite su gestión. El sistema *bck* permite las siguientes acciones:

- Añadir fichero a la copia (*bcklink*)

- Ver los identificadores y ficheros asociados (*bckview*)
- Modificar su id (*bckmv*)
- Restaurar el fichero (*bckrestore*)
- Eliminar el fichero de la copia (*bckrm*)

El funcionamiento de cada una de estas acciones se detalla con la descripción del comando asociado, que se encuentra entre paréntesis en la relación anterior.

## Estructura de Archivos

La siguiente estructura de directorios soporta la funcionalidad de *bck*:



Esta estructura es utilizada por los comandos de utilidad para gestionar la copia. El fichero *mypaths.txt* contiene en cada línea una pareja *id,ruta* (separada por una coma) donde *id* es el identificador del fichero en *bck* mientras que *ruta* indica su ubicación original.

El directorio *files* contiene los archivos que se han incluido en *bck*, pero se sustituye el nombre del fichero por el del *id* asignado. El uso del identificador se justifica para poder gestionar archivos con el mismo nombre, a los que se les asigna un *id* distinto.

La forma en que funciona *bck* es sustituyendo el archivo original por un enlace simbólico que apunta a la copia realizada (el archivo dentro de la carpeta *files* nombrado por su *id*).

Por ejemplo, si se tiene el siguiente fichero:

```
~/dir/test/$ ls -l
-rw-rw-r-- 1 user 10 feb 10 18:52 file1.txt
```

y se añade el fichero al sistema con el identificador *f1*, el mismo directorio ahora tendría un enlace simbólico hacia el directorio de *bck* correspondiente:

```
lrwxrwxrwx 1 user 52 feb 10 20:10 file1.txt -> /home/user/bckdir/files/f1
```

Con respecto a *bck*, el fichero *mypaths.txt* contendría lo siguiente:

```
f1,/home//dir/test/file1.txt
```

y el directorio *files* tendría el siguiente contenido:

```
~/bckdir/files/$ ls -l
-rw-rw-r-- 1 user 13 mar 10 18:52 f1
```

La funcionalidad completa se describe en los comandos correspondientes.

## Consideraciones

- Cuando se ejecuta cualquiera de los comandos de utilidad tiene que crear esta estructura básica, en caso de que no exista

## Programas de Utilidad

Los comandos de utilidad de *bck* son los siguientes:

### bcklink

- Sintaxis

\$ bcklink id path

- Funcionalidad
  - Copia el fichero con ruta *path* con el identificador *id* en *bck*
  - Crea un enlace simbólico que apunta desde *path* al archivo correspondiente *id* en *bck*
  - Actualiza el fichero *mypaths.txt* añadiendo la entrada siguiente

id,path

- Consideraciones
  - Si el identificador ya existe en *bck* muestra un mensaje indicándolo y no realiza ninguna acción

### bckview

- Sintaxis

\$ bckview

- Funcionalidad
  - Muestra el contenido del fichero *mypaths.txt*

### bckmv

- Sintaxis

\$ bckmv old\_id new\_id

- Funcionalidad
  - Cambia el el identificador *old\_id* por el identificador *new\_id*
  - Si *old\_id* no existe no se realiza ninguna acción, mostrando el mensaje pertinente
  - Si *new\_id* ya existe no se realiza ninguna acción, mostrando el mensaje pertinente
  - El cambio tiene que tener efecto en:
    - \* El contenido del fichero *mypaths.txt*, actualizando el *id*

- \* El fichero *id* en el directorio *files*
- \* El enlace simbólico del sistema que apuntaba a *old\_id* tiene que pasar a apuntar a *new\_id*

### **bckrestore**

\$ **bckrestore** id

- Funcionalidad
  - Crea el enlace simbólico que apunta a *id*
  - Si *id* no existe no se realiza ninguna acción, mostrando el mensaje pertinente

### **bckrm**

\$ **bckrm** id

- Funcionalidad
  - Elimina el identificador de *bck*
  - Si *id* no existe no se realiza ninguna acción, mostrando el mensaje pertinente
  - El enlace simbólico que apunta al archivo *id* en *files* es sustituido por el fichero original, del que se había realizado la copia

### **Consideraciones**

- Para todos los comandos, si estos no tienen los comandos esperados (en número), se muestra un mensaje indicándolo y no se realiza ninguna acción

## **Entrega**

La entrega de la práctica tiene como fecha límite el 14 de Abril de 2023 a las 23:59 horas.

Los artefactos a entregar serán los ficheros ejecutables que constituyen la herramienta *bck*.