

Universidad Tecnológica de Panamá
Sistemas Operativos I
Experiencia Práctica 3

Prof. Aris Castillo de Valencia

Nombre: Gabriel Grimaldo

Cédula: 8-949-56

Grupo: 1IF131

Objetivo:

Probar y distinguir distintos comandos para realizar las siguientes actividades en el sistema operativo:

- Empaquetar varios archivos para transportarlos más fácilmente.
- Comprimir y descomprimir archivos.
- Instalar aplicaciones

Procedimiento:

Lea cuidadosamente la guía; pruebe cada uno de los comandos listados prestando especial atención a los resultados obtenidos y a las variantes que le ofrecen las opciones de los comandos. Ponga en práctica los comandos aprendidos haciendo los ejercicios sugeridos. Llene la autoevaluación y retroalimentación y súbala a la plataforma Moodle.

¿Se puede poner varios archivos en uno solo, pero manteniéndolos independientes?

Linux ofrece la facilidad de empaquetar archivos con el comando **tar**. Por ejemplo: `tar cf file.tar file1 file2`

Resultado: copia en file.tar los archivos file1 y file2

```
ghr@ghr-VirtualBox:~$ tar cf lab.tar File1 File2
ghr@ghr-VirtualBox:~$
```



Nota: se puede tratar de más de dos archivos.

Tar es un tipo de archivo tipo bitstream (flujo de bits) y también el programa usado para manejar dichos archivos. Este tipo de archivo fue creado para escribir directamente de forma secuencial en los dispositivos de E/S para realizar backups. Actualmente, se utiliza para agrupar un conjunto de archivos en uno solo para su distribución más expedita. Mantiene la información del sistema de archivos, incluyendo permisos de grupo, de usuario, fechas y estructuras de directorios.

¿Cómo comprimo archivos?

Se puede comprimir un archivo, es decir, reducir el tamaño del archivo con el comando **gzip**. Así:
`gzip file`

Resultado: se comprime el archivo file. El archivo comprimido tiene la extensión .gz

```
ghr@ghr-VirtualBox:~$ gzip lab
```



File.gz

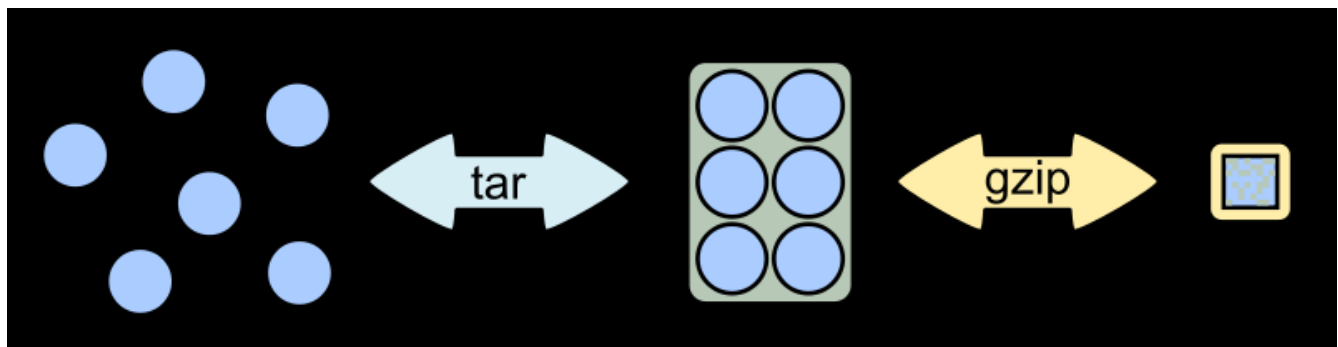
También se puede comprimir un archivo tar con Gzip. Para ello, el comando debe llevar la opción z, así: tar czf file.tar.gz file1 file2

```
ghr@ghr-VirtualBox:~$ tar czf file.tar.gz File1 File2
ghr@ghr-VirtualBox:~$ ls
archivo1.c  Downloads  File2      lb.SO      prueba.c   vim
archivo.c   ejecutable File.gz    Music      Public
Desktop     ejecutable1 file.tar.gz Pictures   Templates
Documents   File1      lab.tar    procesoshoy.txtps Videos
ghr@ghr-VirtualBox:~$
```



file.tar.gz

Como se aprecia en la figura siguiente, tar empaqueta los archivos, mientras que Gzip comprime el archivo de manera que sea más fácil enviarlo por la red o transportarlo en algún dispositivo de almacenamiento externo.



Fuente: [2] Tar File Format

También se puede obtener un resultado similar, pero con el formato Bzip2, en este caso la opción del comando es j; así: tar cjf file.tar.bz2 file1 file2

```
ghr@ghr-VirtualBox:~$ tar cjf file.tar.bz2 File1 File2
ghr@ghr-VirtualBox:~$ ls
archivo1.c  Downloads  File2      lab.tar    procesoshoj.txtps  Videos
archivo.c   ejecutable File.gz    lb.SO      prueba.c          vim
Desktop     ejecutable1 file.tar.bz2 Music      Public
Documents   File1      file.tar.gz Pictures   Templates
ghr@ghr-VirtualBox:~$
```

¿Cómo descomprimo archivos?

Para descomprimir se utiliza también el comando gzip con la opción -d. Así: `gzip -d file.gz`

Resultado: descomprime el archivo file.gz en el archivo file

Si el archivo es tar, entonces se utilizaría el comando **tar** para desempaquetar el archivo.

Ejemplo 1: `tar xf file.tar`

Resultado: se sacan todos los archivos contenidos en el archivo file.tar. Éstos quedarán ubicados en el directorio actual.

```
ghr@ghr-VirtualBox:~$ tar xf lab.tar
ghr@ghr-VirtualBox:~$
```



Ejemplo 2: `tar xzf file.tar.gz`

Resultado: descomprime el archivo file.tar.gz. Todos los archivos quedarán ubicados en el directorio actual.

```
ghr@ghr-VirtualBox:~$ tar xzf file.tar.gz
ghr@ghr-VirtualBox:~$
```

Ejemplo 3: `tar xjf file.tar.bz2`

¿Cuál es el resultado?

El archivo se descomprime y los deja en la ubicados en el directorio actual.

```
ghr@ghr-VirtualBox:~$ tar xjf file.tar.bz2
ghr@ghr-VirtualBox:~$
```

1. Comprima en un archivo tar con Bzip2 tres archivos cualesquiera. Borre los tres archivos. Verifique la acción. Ahora descomprima el archivo tar. Verifique la acción. Escriba la secuencia de comandos tal como los ejecutó.
 - Se comprime los tres archivos en un bzip2


```
ghr@ghr-VirtualBox:~$ tar cjf Files.tar.bz2 doc1.txt doc2.txt doc3.txt
ghr@ghr-VirtualBox:~$
```
 - Se borran los tres archivos

```

Desktop  doc2.txt  Documents  Files.tar.bz2  Music  Public  Videos
doc1.txt  doc3.txt  Downloads  lb.SO         Pictures  Templates  vim
ghr@ghr-VirtualBox:~$ rm doc1.txt doc2.txt doc3.txt
ghr@ghr-VirtualBox:~$ ls
Desktop  Downloads  lb.SO  Pictures  Templates  vim
Documents  Files.tar.bz2  Music  Public  Videos
ghr@ghr-VirtualBox:~$

```

- Se descomprime el Bzip 2 y reaparecen los archivos

```

ghr@ghr-VirtualBox:~$ tar xjf Files.tar.bz2
ghr@ghr-VirtualBox:~$ ls
Desktop  doc2.txt  Documents  Files.tar.bz2  Music  Public  Videos
doc1.txt  doc3.txt  Downloads  lb.SO         Pictures  Templates  vim
ghr@ghr-VirtualBox:~$

```

¿Cómo puedo instalar aplicaciones en mi sistema?

Dependiendo de la distribución de GNU/Linux que se utiliza, así es la forma de instalar paquetes. A continuación, se describen algunas formas útiles.

Compilación e instalación desde archivos fuente.

Si el software que desea instalar está en forma de fuente, usted debe desempaquetarlo para luego instalarlo. Estos archivos generalmente están en alguno de estos formatos: .tar, .bz2, o tar.gz o .zip.

El procedimiento sería:

1. Descomprimir el archivo, utilizando alguna de las formas siguientes:

<code>tar -zxvf <filename></code>	Si el archivo es tar.gz
<code>tar -jxvf <filename></code>	Si el archivo es tar.bz2
<code>unzip <filename></code>	Si el archivo es .zip

2. Ahora se tiene un directorio que contiene los archivos fuentes. Entre a este directorio y revise si hay algún archivo README o INSTALL, que le de instrucciones precisas sobre la instalación.
3. Ejecute el archivo de pre-configuración, con el comando **./configure**
4. Ahora compile el software, con el comando **make**
5. Finalmente instale el software, con el comando **make install**

Instalación usando RPM Package Management

Esto funciona para RedHat of Fedora. En este caso ya los paquetes están compilados y vienen en un archivo extensión .rpm. Para instalar el paquete sólo se debe ejecutar el comando como sigue usando el usuario root

```
rpm -i <filename.rpm>
```

la opción -i significa instalar

Instalación con apt-get

Esta opción funciona para sistemas Debian o sus derivados. APT significa “application packing tool,” es decir una herramienta de empaquetamiento. Esta herramienta realiza todo el proceso de bajar los archivos y realizar las dependencias. La herramienta APT también tiene una interfaz gráfica llamada

Synaptic.

Para instalar paquetes binarios, ejecute el comando así `apt-get install <package-name>`

Para instalar paquetes fuentes, ejecute el comando así `apt-get source -b <package-name>`

Para dependencias, ejecute el comando así `apt-get build-dep <package-name>`

Para saber sobre el paquete y sus dependencias, ejecute el comando así `apt-cache show <package-name>`

Instalación con yum

Esta forma funciona con Fedora. Para instalar ejecute el comando **yum install** y para actualizar el paquete, ejecute el comando **yum update**.

¿Cómo desinstalo aplicaciones?

Para desinstalar un archivo RPM se debe ejecutar el comando **rpm -e <filename.rpm>**, donde la opción -e significa erase o borrar.

Para desinstalar paquetes con APT, ejecute el comando **apt-get remove <package_name>** o `apt-get remove - - purge <package_name>`

Para desinstalar paquetes con yum, ejecute el comando **yum remove**.

Ejercicio adicional.

Ahora haga la prueba instalando Python en Linux. Haga las pruebas de acuerdo con su sistema. Documente la experiencia.

```
ghr@ghr-VirtualBox:~$ sudo apt-get install python3.8
Reading package lists... Done
Building dependency tree
Reading state information... Done
python3.8 is already the newest version (3.8.2-1ubuntu1.2).
python3.8 set to manually installed.
The following packages were automatically installed and are no longer required:
  apg apport-symptoms aptdaemon-data avahi-utils blt checkpolicy cockpit
  cockpit-bridge cockpit-dashboard cockpit-networkmanager cockpit-packagekit
  cockpit-system cockpit-ws cups-pk-helper dc distro-info-data docbook-xml
  ethtool fonts-glyphicons-halflings fonts-lyx fprintd
  gir1.2-accountsservice-1.0 gir1.2-atspi-2.0 gir1.2-dbusmenu-glib-0.4
  gir1.2-dee-1.0 gir1.2-gck-1 gir1.2-gcr-3 gir1.2-gdm-1.0 gir1.2-geoclue-2.0
  gir1.2-gnomebluetooth-1.0 gir1.2-goa-1.0 gir1.2-graphene-1.0
  gir1.2-gst-plugins-base-1.0 gir1.2-gstreamer-1.0 gir1.2-gtksource-4
  gir1.2-gudev-1.0 gir1.2-gweather-3.0 gir1.2-ibus-1.0
  gir1.2-javascriptcoregtk-4.0 gir1.2-json-1.0 gir1.2-mutter-6 gir1.2-nm-1.0
ghr@ghr-VirtualBox:~$ python3 --version
Python 3.8.2
```

Se ejecuta el comando `sudo apt-get install python3.8` y se instala inmediatamente.

Luego se rectifica para saber que se instalo la version correcta esto se realiza con el comando `python3 --version`.

Si toda esta informacion concide quiere decir que se instalo la version deseada.

Retroalimentación y autoevaluación.

1. Documente cada uno de los puntos, preguntas y ejercicios de la guía.
2. Busque 5 comandos relacionados con los discutidos en esta guía. Pruébelos. Describa sus usos y escriba ejemplos específicos completos, incluyendo la sintaxis y opciones utilizadas.
 1. APT (Advance Packagin Tool): Es una herramienta de empaquetado de aplicaciones creada por el proyecto Debian, para simplificar la instalación y desinstalación de programas en Linux.
 1. Ejemplo: `sudo apt-get install firefox`
 2. `lha -v` : Es utilizado para ver los contenidos dentro de un archivo .lha
 1. Ejemplo: `lha -v archivo.lha`
 3. `Rar -a` : Se utiliza para comprimir en formato rar
 1. Ejemplo: `rar -a archivo.rar /mayo/archivos`
 4. `zoo -x`: Se utiliza para descomprimir archivos de formato zoo
 1. Ejemplo: `zoo -x archivo.zoo`
 5. `Zip`: Se utiliza para comprimir en formato zip.
 1. Ejemplo: `zip archivo.zip /mayo/archivos`
3. ¿En qué situaciones específicas considera que serían útiles los comandos utilizados?
Serian útiles para comprimir una gran cantidad de documentos de forma inmediata, también para instalar aplicaciones o removerlas.
4. ¿Qué dificultades encontró durante el desarrollo del laboratorio?
La única dificultad que encontré fue buscar cual era la última versión de Python disponible, debido a que en algunos blogs decían que era la 3.6 y otros la 3.8.
5. ¿Qué mejoraría de esta experiencia de laboratorio?
De este laboratorio no mejoraría nada porque considero que me ayudo a comprender muy bien el uso de estos comandos.

Referencias:

1. Kernighan, B. y Pike, R. El Entorno de programación Unix. Prentice Hall.
2. Tar File format: http://en.wikipedia.org/wiki/Tar_%28file_format%29
3. Begginer Linux Tutorial: <http://beginnerlinuxtutorial.com/help-tutorial/basic-linux-commands/>
4. Different ways to install softwares in GNU/Linux: <http://forums.techarena.in/guides-tutorials/1277676.htm>
5. Examples of tar command:
<http://www.mppmu.mpg.de/Introduction/general/subsubsection2.5.8.2.1.html>