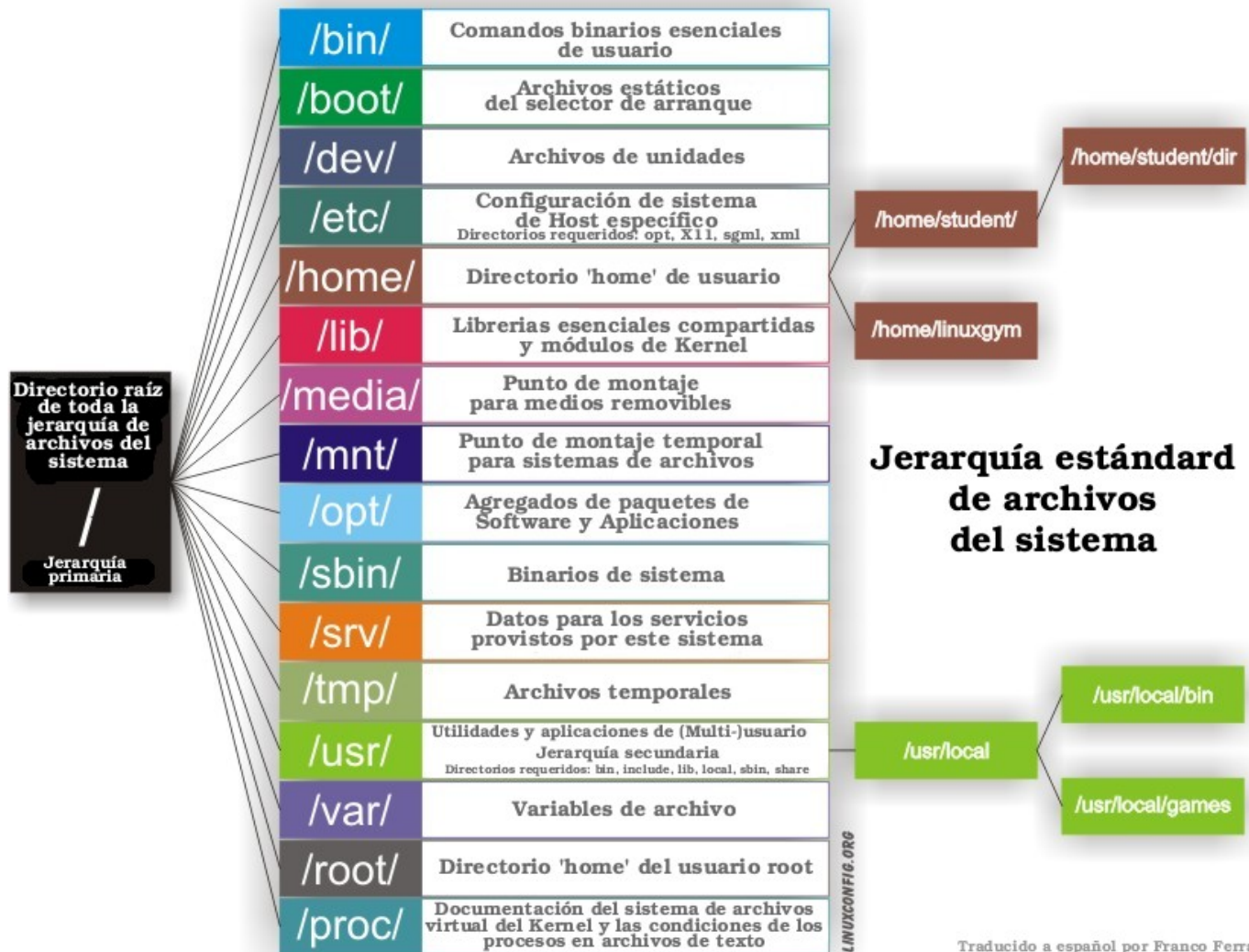


INTRODUCCION LINUX

Linux- Estructura de directorios



Comandos básicos para el manejo de archivos y directorios

Comando/Sintaxis	Descripción	Ejemplos
<code>cat <i>fich1</i> [...<i>fichN</i>]</code>	Concatena y muestra un archivos	<code>cat /etc/passwd</code>
	archivos	<code>cat dict1 dict2 dict</code>
<code>cd [<i>dir</i>]</code>	Cambia de directorio	<code>cd /tmp</code>
<code>chmod <i>permisos fich</i></code>	Cambia los permisos de un archivo	<code>chmod +x miscript</code>
<code>chown <i>usuario:grupo fich</i></code>	Cambia el dueño un archivo	<code>chown nobody miscript</code>
<code>cp <i>fich1...fichN dir</i></code>	Copia archivos	<code>cp foo foo.backup</code>
<code>diff [-e]<i>arch1 arch2</i></code>	Encuentra diferencia entre archivos	<code>diff foo.c newfoo.c</code>
<code>du [-sabr] <i>fich</i></code>	Reporta el tamaño del directorio	<code>du -s /home/</code>
<code>file <i>arch</i></code>	Muestra el tipo de un archivo	<code>file arc_desconocido</code>

Comandos básicos para el manejo de archivos y directorios

<code>find dir test acción</code>	Encuentra archivos.	<code>find . -name ``.bak" -print</code>
<code>grep [-cilmv] expr archivos</code>	Busca patrones en archivos	<code>grep mike /etc/passwd</code>
<code>head -count fich</code>	Muestra el inicio de un archivo	<code>head prog1.c</code>
<code>mkdir dir</code>	Crea un directorio.	<code>mkdir temp</code>
<code>mv fich1 ...fichN dir</code>	Mueve un archivo(s) a un directorio	<code>mv a.out prog1</code>
<code>mv fich1 fich2</code>	Renombra un archivo.	<code>mv .c prog_dir</code>
<code>less / more fich(s)</code>	Visualiza página a página un archivo.	<code>more muy_largo.c</code>
	less acepta comandos vi.	<code>less muy_largo.c</code>

Comandos básicos para el manejo de archivos y directorios

<code>ls</code>	Lista el contenido del directorio	<code>ls -l /usr/bin</code>
<code>pwd</code>	Muestra la ruta del directorio actual	<code>Pwd</code>
<code>rm <i>fich</i></code>	Borra un fichero.	<code>rm foo.c</code>
<code>rm -r <i>dir</i></code>	Borra un todo un directorio	<code>rm -rf prog_dir</code>
<code>rmdir <i>dir</i></code>	Borra un directorio vacío	<code>rmdir prog_dir</code>
<code>tail -count <i>fich</i></code>	Muestra el final de un archivo	<code>tail prog1.c</code>
<code>vi <i>fich</i></code>	Edita un archivo.	<code>vi .profile</code>

Creación y eliminación de cuentas de usuario

- Linux es un s.o. diseñado para ser utilizado por múltiples usuarios. No es conveniente realizar el trabajo diario desde la cuenta de **root** la cual sólo debe utilizarse para la administración del sistema .
- Una cuenta de **usuario** contiene las restricciones necesarias para impedir que se ejecuten mandatos que puedan dañar el sistema, se altere accidentalmente la configuración del sistema, los servicios que trabajan en el trasfondo, los permisos y ubicación de los archivos y directorios de sistema, etc.
- Los comandos básicos son: `useradd`, `passwd`,
 `useradd nombre-usuario`
 `passwd nombre-usuario`

Editores de texto – vim

- vim es uno de los editores de texto más poderosos. Útil a fin de facilitar la edición de archivos de texto simple (archivos de configuración)
- El editor vim tiene dos modos básicos de operación, modo COMANDO y modo de EDICION. Por defecto, vim se encuentra en el modo comando.
- Para poder insertar texto se usa la tecla: *i*
Y para regresar a modo comando la tecla: *ESC*
Para ejecutar al editor e iniciar una sesion en el mismo se debe
teclear: *vim nombre_archivo*

Editores de texto – vim

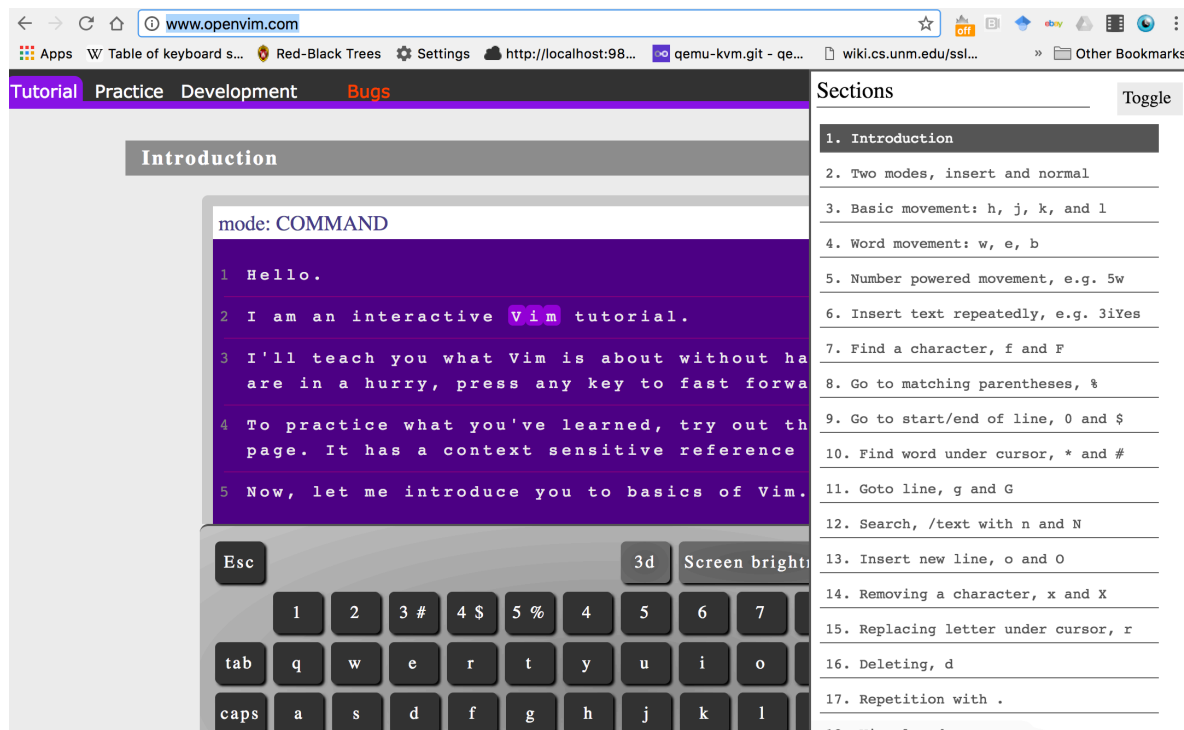
Combinación	Resultado
I (i + SHIFT)	Inicia insertar texto al inicio de la línea donde se encuentra el cursor
A (a + SHIFT)	Inicia insertar texto al final de la línea donde se encuentra el cursor.
o	Abre una nueva línea e inicia insertar texto en la nueva línea.
x	Elimina el carácter que esté sobre el cursor.
dd	Elimina la línea actual donde se encuentre el cursor.
D	Elimina desde la posición actual del cursor hasta el final de la misma línea donde se encuentra el cursor.
dG	Elimina todo hasta el final del fichero.
:q	Aparece si no hubo cambios en el ficheros.
:q!	Aparece descartando los cambios en el fichero.

Editores de texto – vim

<code>:w</code>	Guarda el fichero sin salir.
<code>:wq</code>	Guarda el fichero y sale de vi.
<code>:x</code>	Lo mismo que <code>:wq</code>
<code>:saveas /lo/que/sea</code>	Guarda el fichero como otro fichero donde sea necesario.
<code>:wq! ++enc=utf8</code>	Codifica el fichero en UTF-8.
<code>:u</code>	Deshacer cambios
<code>:red</code>	Rehacer cambios.
<code>:/cadena caracteres</code> de	Búsqueda de cadenas de caracteres.
<code>:nohl</code>	Cancelar el resaltado de resultados de Búsqueda.

Tutorial Vim online

Disponible en: <http://www.openvim.com/>



Gestión de permisos

- La asignación de permisos de acceso (de lectura, escritura y ejecución) pueden asignarse a través de modos, que son combinaciones de números de tres dígitos (usuario, grupo y resto del mundo) y el mandato **chmod**.
- El esquema de notación simbólica se compone de 10 caracteres, donde el primer carácter indica el tipo de archivo:

Valor	Descripción
-	Denota un fichero regular.
d	Denota un directorio.
b	Denota un fichero especial de dispositivos de bloque.
c	Denota un fichero de carácter especial
l	Denota un enlace simbólico.
p	Denota una tubería nombrada (FIFO)
s	Denota un zócalo de dominio (socket)

chmod

Gestión de permisos para el sistema de archivos

- Cada clase de permisos es representada por un conjunto de tres caracteres. El primer conjunto de caracteres representa la clase del **usuario**, el segundo conjunto la clase del **grupo** y el tercer conjunto la clase de «**otros**». Cada uno de los tres caracteres representa permisos de lectura, escritura y ejecución, respectivamente y en ese orden.

Permisos	Descripción
d rw xr - xr - x	Directorio con permiso 755.
c rw- rw - r --	Fichero de carácter especial con permiso 664.
s rw xrwxr - x	Zócalo con permiso 775.
p rw- rw - r --	Tubería (FIFO) con permiso 664.

chmod: Permisos Numéricos

#	Permission	rwX
7	read, write and execute	rwX
6	read and write	rw-
5	read and execute	r-X
4	read only	r--
3	write and execute	-wX
2	write only	-w-
1	execute only	--X
0	none	---

Uso de la herramienta apt-get (Ubuntu)

- apt-get es una herramienta que permite descargar y actualizar paquetes al sistema.
- Se requiere tener una conexión a Internet y un buen ancho de banda.
- Comandos básicos.

Para instalar: *apt-get install nombre-paquete*

Para desinstalar. *apt-get remove nombre-paquete*

Actualizar lista de paquetes disponibles: *apt-get update*

- Mas comandos en:
<https://help.ubuntu.com/community/AptGet/Howto>

Paquetes con extension .deb

- Otra forma de instalar aplicaciones en el sistema es por medio de los paquetes ya preparados para ser instalados y con extensión .deb.

- Para instalar

```
sudo dpkg -i nombre_del_paquete.deb
```

- Para desinstalar

```
sudo dpkg -r nombre_del_paquete
```

Configuración de parámetros de red

- Dirección de Red, Mascara, Puerta de enlace: Debe modificarse con cualquier editor de textos y verificar que sus parámetros de red sean los correctos. En Ubuntu 16.04, el archivo está localizado en la ruta **/etc/network/interfaces**

```
hadoop@node-master:~$ cat /etc/network/interfaces
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback
hadoop@node-master:~$
```

- Ejemplo configuración ip estática
auto enp0s25
iface enp0s25 inet static
address 192.168.0.16
netmask 255.255.255.0
gateway 192.168.0.1
dns-nameservers 8.8.4.4 8.8.8.8

Configuración de parámetros de red

- Servidores de nombres: Debe modificarse con un editor de textos **/etc/resolv.conf** y deben establecerse en éste los servidores de resolución de nombres de dominio (DNS).
- Ejemplo:

```
nameserver 8.8.8.8
```
- Después de haber configurado todos los parámetros de red deseados, sólo deberá ser reiniciado el servicio de red, ejecutando lo siguiente:

```
service networking restart
```
- *ifconfig* permite examinar la información de configuración de las interfaces de red

Tarea (Opcional)

Realizar el taller de Linux disponible en classroom