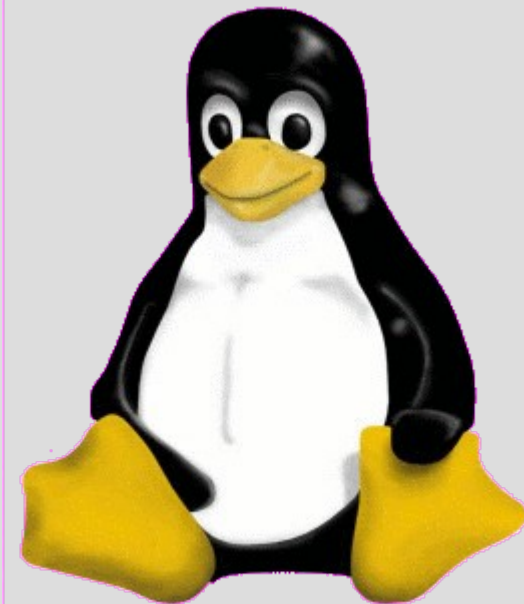


Introducción a Linux

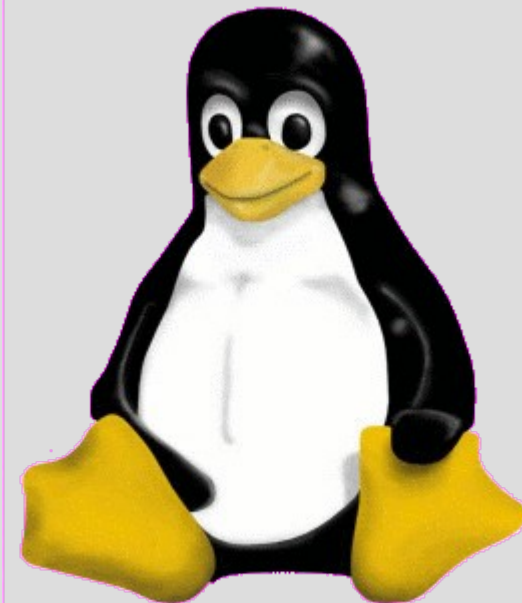
El regreso a los años 70: la consola (I)



Introducción a Linux

El regreso a los años 70: la consola (I)

- ¿Qué es y cómo acceder a una consola?
- Algunos comandos para moverse
- Sintaxis general de parámetros en BASH
- Encadenamiento de comandos
- Otros comandos GNU
- Editores de textos



Introducción a Linux

El regreso a los años 70: la consola (I)

¿Qué es y cómo acceder a una consola?

Una consola es una terminal virtual de texto (tty), inicialmente tendremos 6 en marcha (podría variar entre distribuciones), accesibles mediante `<CTRL>+<ALT>+<Fn>`

En el entorno X hay emuladores de terminal, en KDE tenemos “konsole”, podemos acceder desde el menú de inicio o desde la opción “Ejecutar comando” :: “konsole”



Introducción a Linux

El regreso a los años 70: la consola (I)

¿Qué es y cómo acceder a una consola?

Desde las consolas reales deberemos loguearnos para identificarnos. A todos los efectos, un login en una consola es igual que en el entorno gráfico. De hecho, el entorno gráfico es una aplicación arrancada desde la consola 7.

Desde las consolas virtuales del entorno gráfico, estaremos autologueados con el mismo usuario que usamos en las X. No es una nueva sesión, sólo una aplicación más de esa misma sesión.



Introducción a Linux

El regreso a los años 70: la consola (I)

¿Qué es y cómo acceder a una consola?

Al acceder a una consola, vemos algo parecido al viejo MSDOS, pero aquí el “intérprete de comandos” (el clásico `command.com`) es un potente interfaz de texto (shell).

Existen varios shells: `ash`, `zsh`, `tsh`, ... Pero el más común por su simplicidad y versatilidad es `BASH`.



Introducción a Linux

El regreso a los años 70: la consola (I)

¿Qué es y cómo acceder a una consola?

En BASH la entrada de comandos se llama *prompt*.

La forma del prompt se puede cambiar, pero inicialmente es algo así:

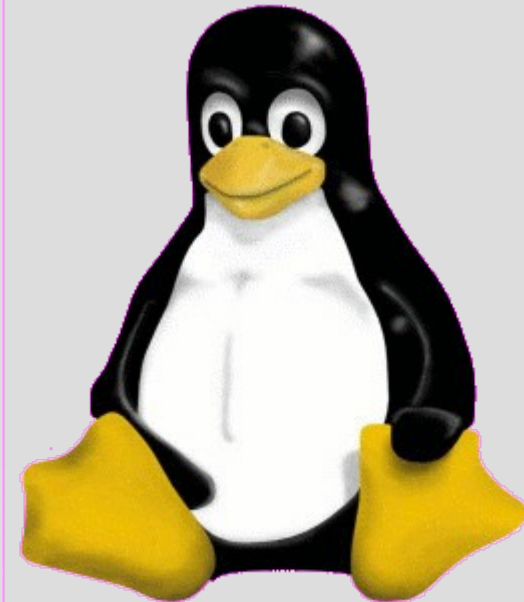
```
[usuario@host cwd]$ |  
[root@host cwd]# |
```



Introducción a Linux

El regreso a los años 70: la consola (I)

Algunos comandos para moverse



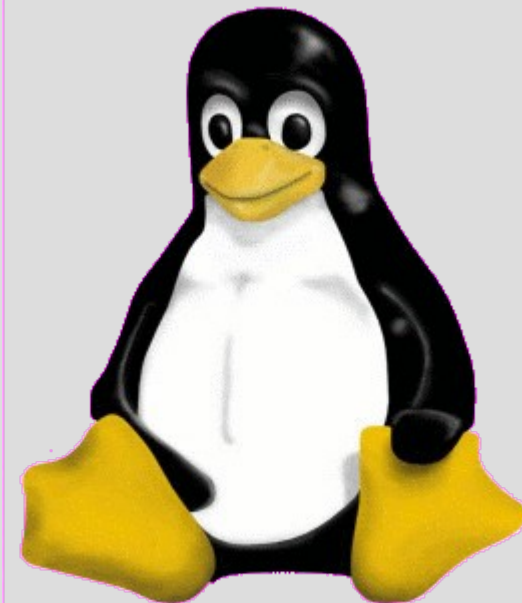
Introducción a Linux

El regreso a los años 70: la consola (I)

Algunos comandos para moverse

Desde el prompt (el cursor de entrada de comandos) de BASH podremos hacer 4 cosas:

- Ejecutar un comando interno.
- Ejecutar un comando externo.
- Consultar el historial de comandos.
- Asignar variables de entorno.



Introducción a Linux

El regreso a los años 70: la consola (I)

Algunos comandos para moverse

Comandos internos más importantes.

- **help** muestra la ayuda de bash, la lista de comandos disponibles y su sintaxis.
- **cd** cambia de directorio
- **pwd** *print working directory*, muestra la ruta actual en la que estamos.
- **alias** nos permite crear nuestros propios comandos de sesión.
- **exit** y **logout** cierran la sesión y la consola vuelve a su estado inicial.



Introducción a Linux

El regreso a los años 70: la consola (I)

Algunos comandos para moverse

Comandos externos GNU: manipulación de archivos

- **ls** lista el contenido del directorio actual.
- **cp** copia un fichero.
- **mv** mueve un fichero a otro directorio o a otro nombre.
- **rm** borra un fichero
- **ln** crea un fichero link o acceso directo
- **mkdir** y **rmdir**, *make directory* y *remove directory*, crea y borra un directorio, resp.



Introducción a Linux

El regreso a los años 70: la consola (I)

Algunos comandos para moverse

Comandos externos GNU: manipulación de archivos

- **cat** muestra el contenido del fichero especificado.
- **touch** crea un fichero nuevo.
- **more y less** paginan el contenido del fichero especificado.
- **man** manual del sistema, muestra las paginas de ayuda del comando especificado.



Introducción a Linux

El regreso a los años 70: la consola (I)

Algunos comandos para moverse

Comandos externos GNU: control de procesos

- **top** muestra dinámicamente los procesos activos del sistema, además de mucha información útil relativa a ellos.
- **ps** similar al anterior, pero no es dinámico.
- **kill** permite detener un proceso, útil para cuando se ha colgado.
- **jobs** y **fg** estos en realidad son comandos internos de BASH, nos permiten ver y reactivar comandos que estén en background.



Introducción a Linux

El regreso a los años 70: la consola (I)

Algunos comandos para moverse

Consultar el historial de comandos ejecutados

- **history** es un comando interno que nos muestra el listado de todos los comandos ejecutados.
- También podemos acceder a ese listado mediante las flechas y navegaremos atrás y adelante en los comandos.
- Y podemos re-ejecutar alguno usando una macro del bash: **!** Si lo escribimos acompañado de un número, ejecutará el comando en esa posición.



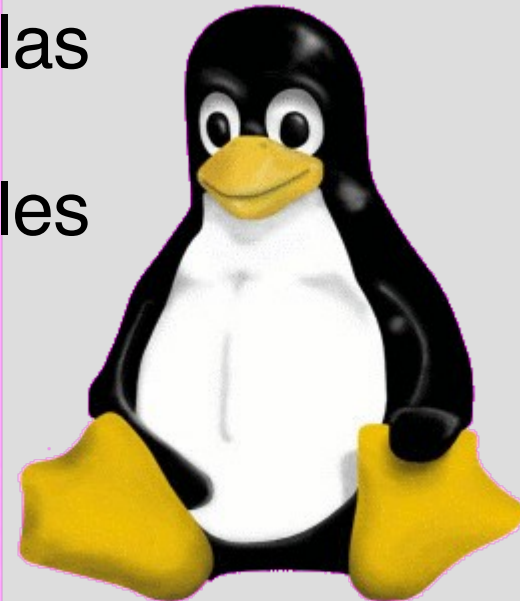
Introducción a Linux

El regreso a los años 70: la consola (I)

Algunos comandos para moverse

Asignación de variables.

- Cada sesión que iniciemos, es un entorno de ejecución diferente y tiene sus propias variables y alias, que inicialmente serán las que estén definidas para el usuario.
- El comando **env** nos muestra las variables disponibles en nuestra sesión.



Introducción a Linux

El regreso a los años 70: la consola (I)

Algunos comandos para moverse

Asignación de variables de entorno.

- Las variables de BASH son tipo string, se crean de manera sencilla:

```
[user@host cwd]$ a=hola
```

- Para acceder a una variable usaremos el carácter “\$” :

```
[user@host cwd]$ echo $a
```

