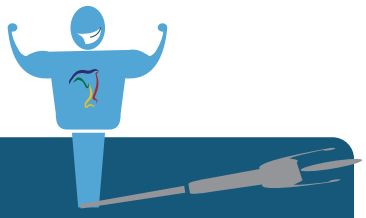
autentia

Avenida de Castilla,1 - Edificio Best Point - Oficina 21B
28830 San Fernando de Henares (Madrid)
tel./fax: +34 91 675 33 06
info@autentia.com - **www.autentia.com**

¿**Qué ofrece** Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
Ese apoyo que siempre quiso tener...

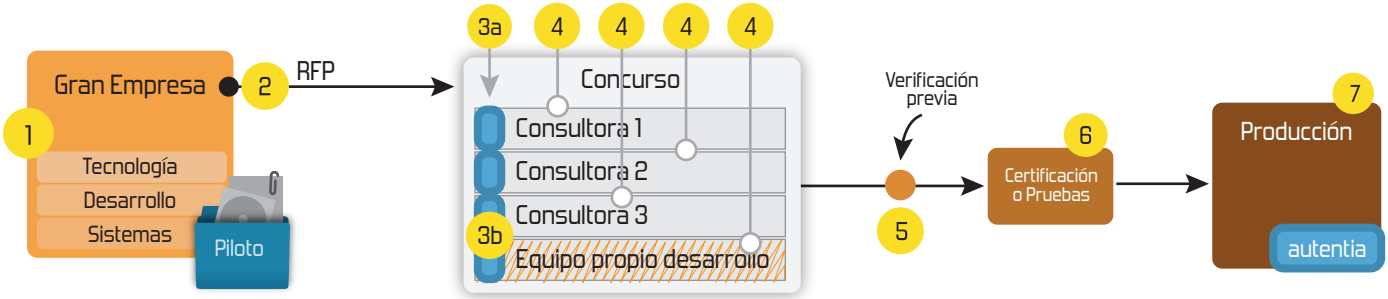
1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces, HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
Gestor de contenidos (Alfresco)
Aplicaciones híbridas

Control de autenticación y acceso (Spring Security)
UDDI
Web Services
Rest Services
Social SSO
SSO (Cas)

JPA-Hibernate, MyBatis
Motor de búsqueda empresarial (Solr)
ETL (Talend)

Dirección de Proyectos Informáticos.
Metodologías ágiles
Patrones de diseño
TDD

Tareas programadas (Quartz)
Gestor documental (Alfresco)
Inversión de control (Spring)

BPM (jBPM o Bonita)
Generación de informes (JasperReport)
ESB (Open ESB)

Compartimos nuestro conocimiento en:
www.adictosaltrabajo.com

Para más información visítenos en:
www.autentia.com

Hosting Patrocinado por enREDados.com

Home | Quienes Somos | Empleo | Foros | Tutoriales | Servicios Gratuitos | Contacte



Tutorial desarrollado por:
Roberto Canales Mora 2003-2005
Creador de AdictosAlTrabajo.com y

Director General de Autentia S.L.

Recuerda que me puedes contratar para echarte una mano:

Desarrollo y arquitectura Java/J2EE
Asesoramiento tecnológico Web
Formación / consultoría integrados en tu proyecto

No te cortes y contacta: 655 99 11 72rcanales@autentia.com.

autentia
real business solutions

Descargar este documento en formato PDF linuxcero.pdf

Base de Datos de Empresas
Pruebe gratis nuestra consulta online. Más de 4 mill de empresas.

Asesoria Empresas
La guía con la información que buscas para ampliar tu formación.

Formacion Empresas
Dirección de Empresas a distancia Máster acreditado a nivel europeo

hotel en cangas de onis
Reuniones empresa hasta 300pax En zona rural con actividades.

Anuncios Goooooogle

Primeros pasos por Linux

Muchas veces me escribe gente diciéndome que los tutoriales en www.adictosaltrabajo.com son muy avanzados.

Todo esto tiene una explicación:

> En mi empresa, www.autentia.com nos dedicamos a dar formación y soporte de desarrollo a otras empresas de desarrollo de Software o a grandes empresas finales. Como podréis entender, no nos contactan para cosas triviales sino que, bien porque su gente cualificada está muy ocupada o porque acaban de llegar a estas tecnologías (Java / J2EE / C++ / Linux / OpenSource), nos hacen encargos avanzados: Arquitectura J2EE, auditorias de código, desarrollo de componentes a medida, subcontratación de pequeños proyectos completos, etc.. Los conocimiento que tenemos que manejar son avanzados ....

Hoy, que he empezado a repasar Linux (también impartimos cursos "básicos"), voy a aprovechar para capturar unas pantallas que os puedan ayudar a repasar conceptos básicos en Linux (para los que lo tengáis oxidado). No tienen muchas explicaciones porque son muy visuales los ejemplos ... simplemente os invito a reproducirlos ...

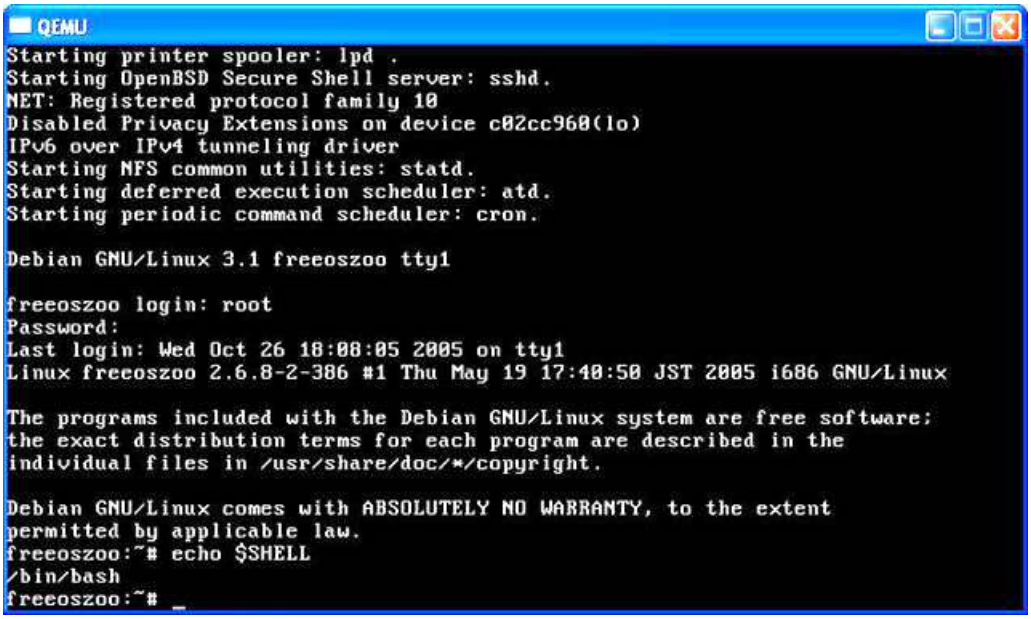
Voy a usar la distribución Debian que viene como imagen en el emulador Qemu (visitad el tutorial anterior http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=qemuwin)..... espero que os sea útil.

Arranque y primeros comando

Una vez logados en Linux, como **root** (ya que lo acabamos de instalar) vamos ha hacer unas primeras comprobaciones

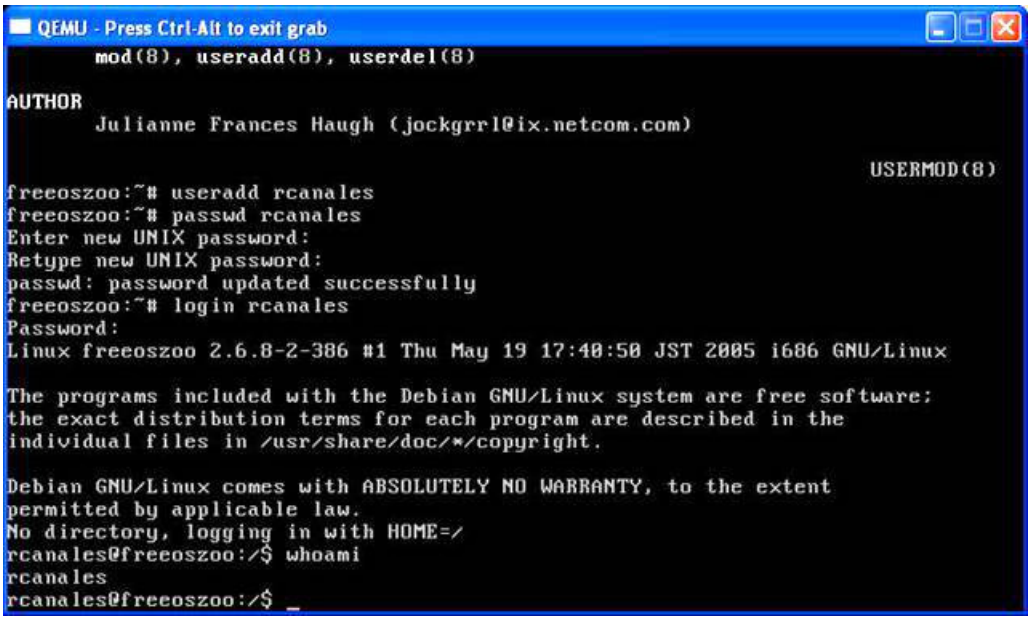
Como sabréis, en Linux hay varios shell y vamos a repasar algunos conceptos en este sentido

Para saber el tipo de shell en uso, **echo $SHELL**

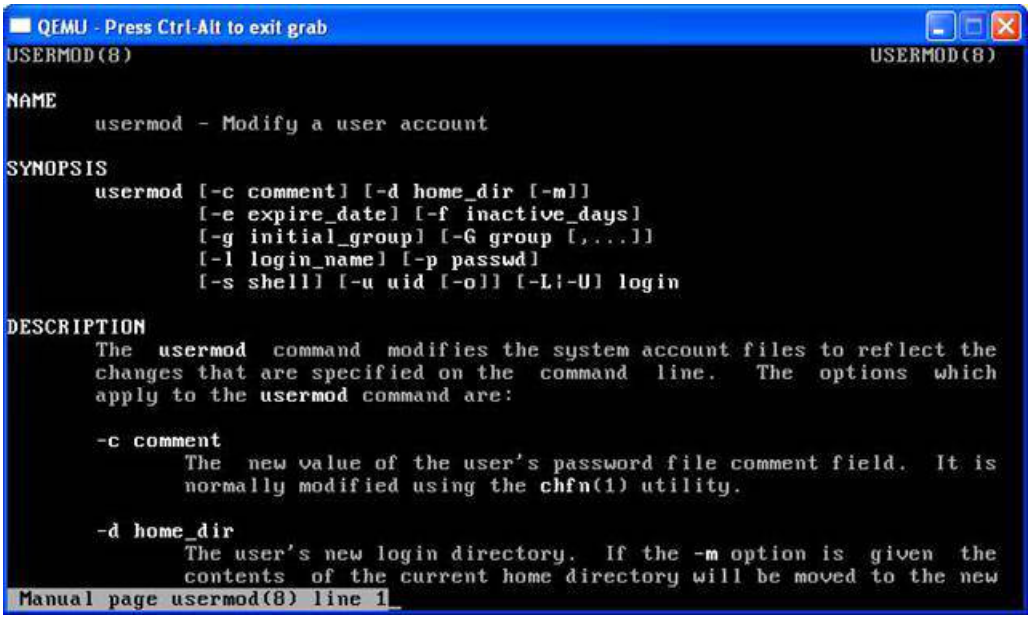


No debemos usar el usuario **root** como norma (por seguridad al igual que no debemos usar el usuario administrador en Windows) por lo que debemos aprender crear uno

Creación de usuarios, usamos los comandos **useradd** y **passwd**

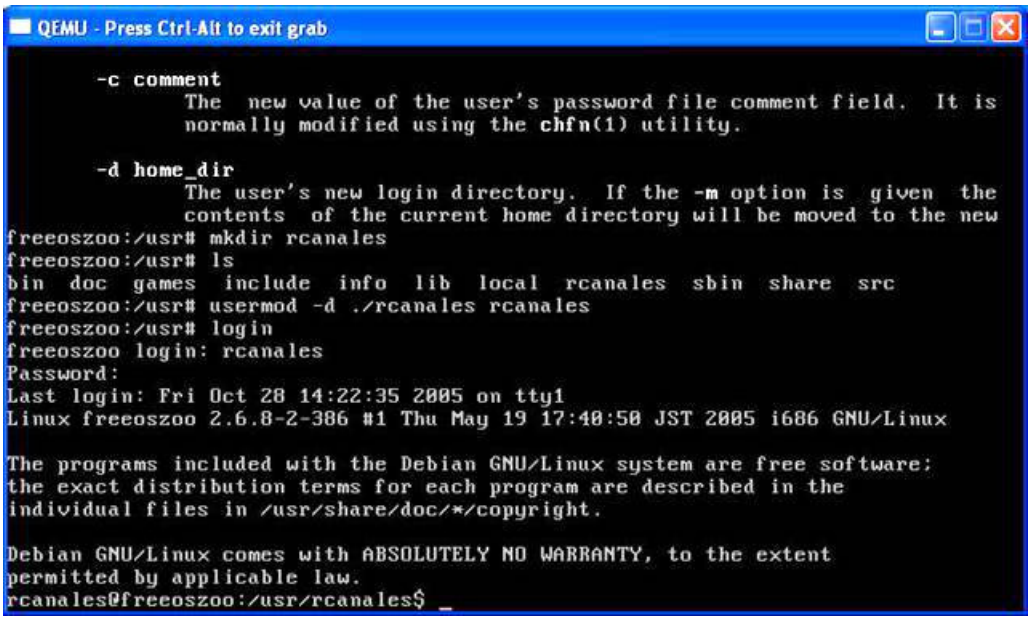


Una vez creado un usuario, para configurarle sus parámetros particulares, usaremos el comando **usermod.** Recordar que con **man usermod nos** aparece la ayuda

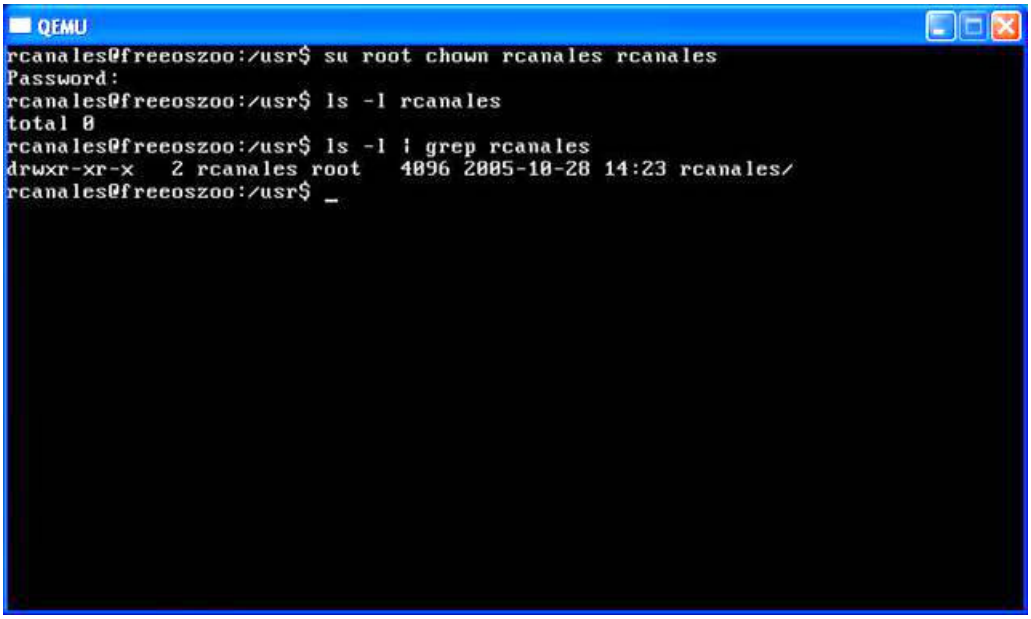


Asignamos el directorio de arranque con la opción **-d.** Aunque os funcione un trayecto relativo, aseguraos de usar un absoluto (el que no lea los texto que pongo, se equivocará)

Lo suyo es que el directorio de usuario colgase de **/home**



No debemos olvidar que debemos asignar la propiedad de ese directorio al usuario: usaremos el comando **chown** (cambiar propietario)

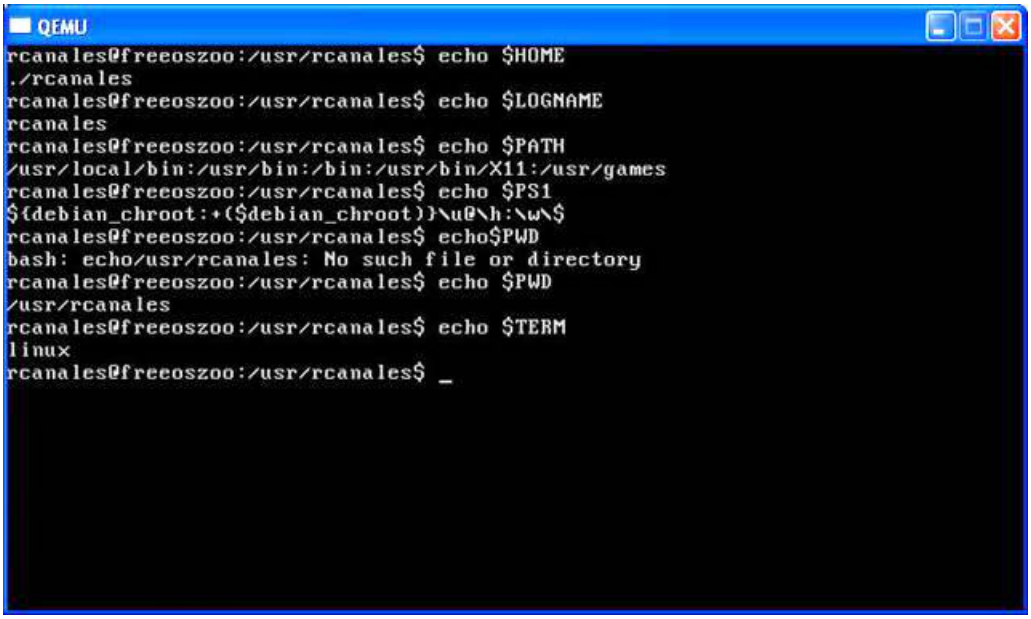


Jugar con el Shell

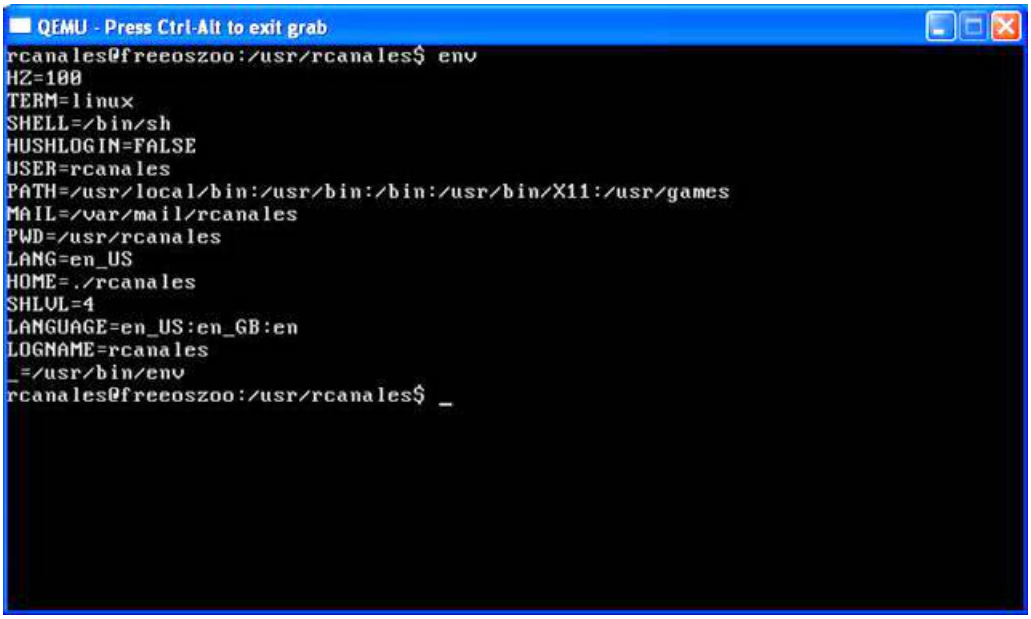
En Linux hay varios shell, recordamos conceptos básicos

- Bourne Shell = sh es el shell original
- bash es el shell por defecto en Linux

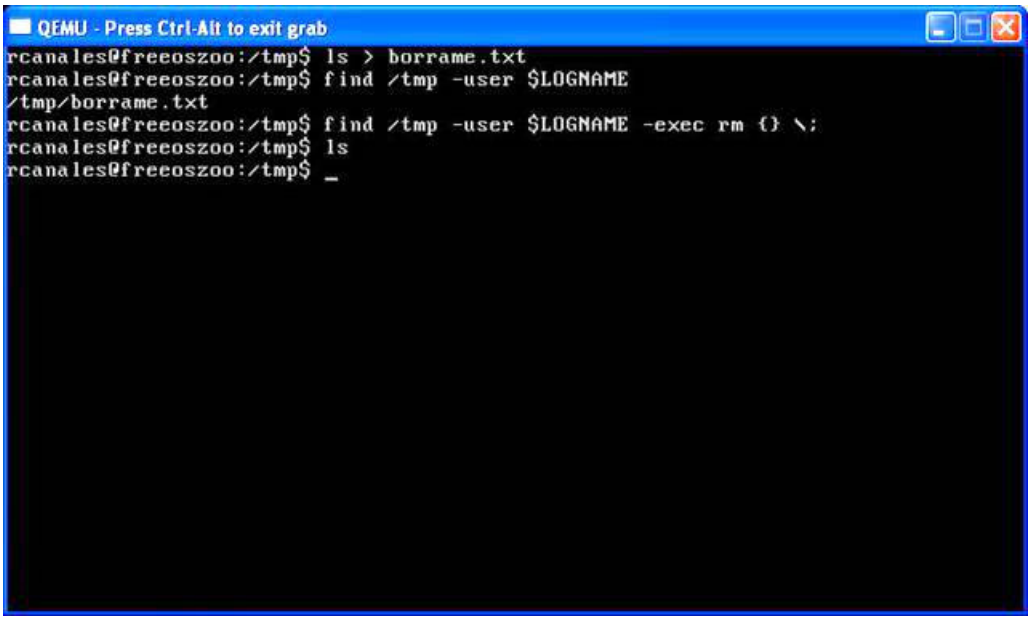
En linux hay muchas variable de entorno pre-establecidas, debemos aprender a manejarnos.



Si ejecutamos el comando **env**, las veremos todas las variables de entorno en nuestro Linux



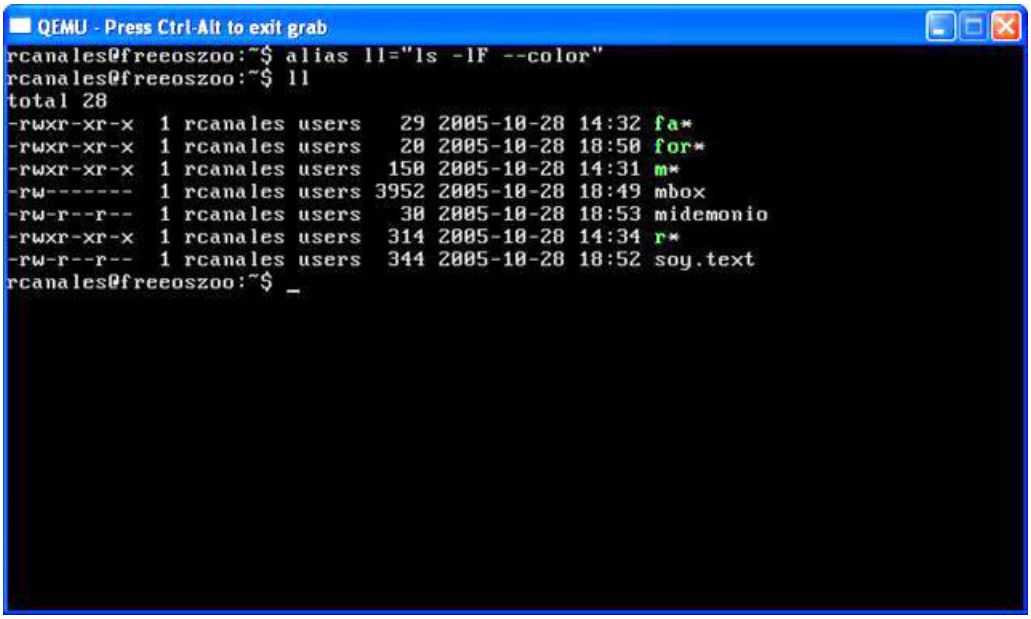
Los comandos más básicos a usar serán **ls, cd, find** ... que combinados con las variables de entorno nos darán mucho juego



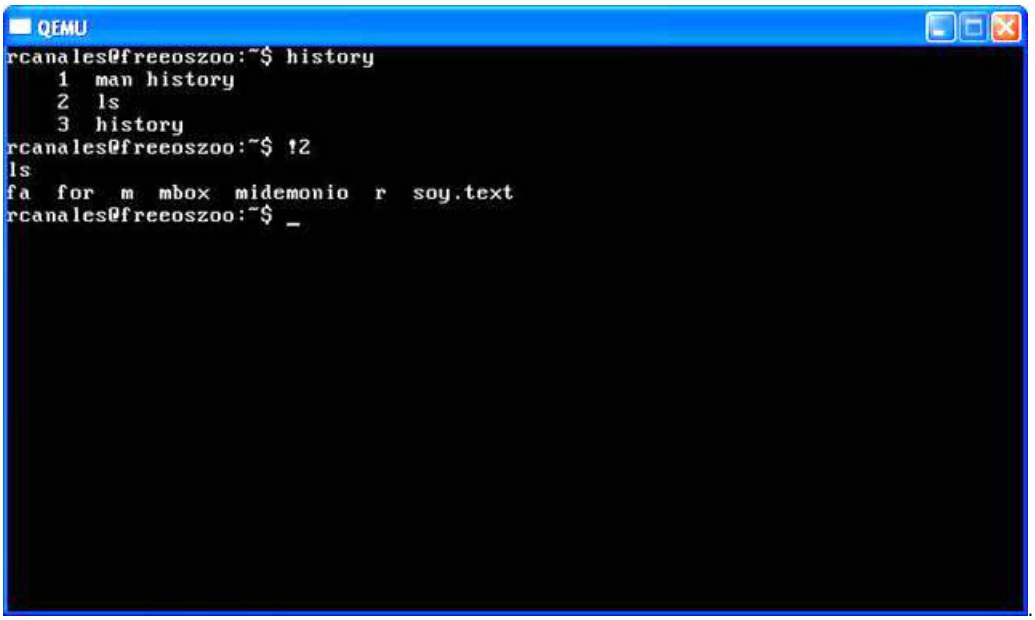
Peculiaridades

Estemos donde estemos, si usamos **cd ~** (AltGr + 4) volveremos al directorio **$HOME**

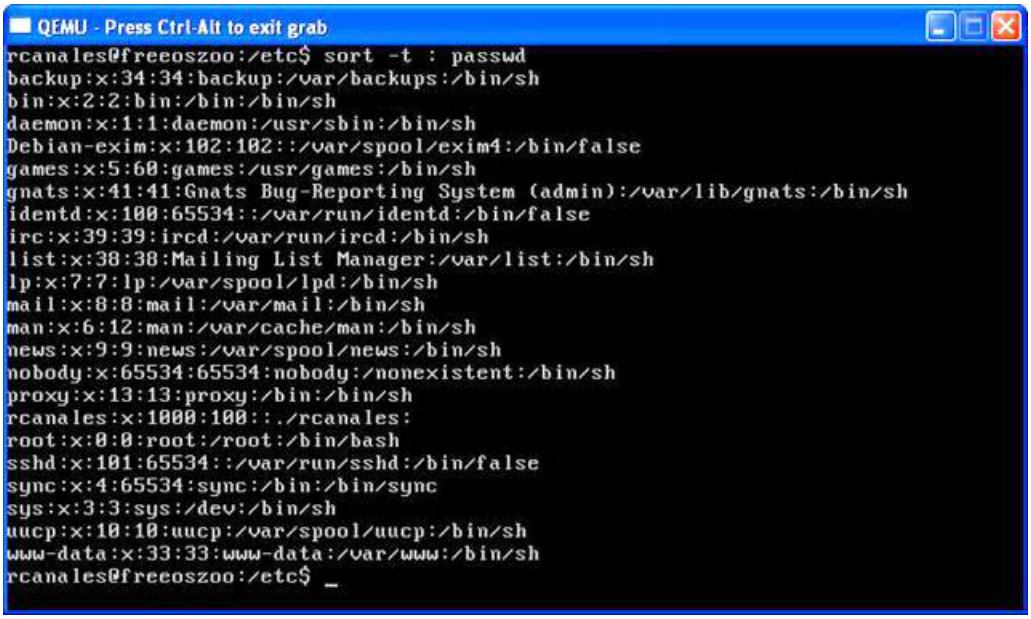
Con el comando **alias** podemos crear nuestros propios comandos. Siempre me ha gustado asignar **alias ll="ls -lF --color".** Así se puede ver de un modo fácil los datos más importantes de un directorio.



Los comandos que vamos escribiendo los podemos recordar ... o bien pulsando la fecha superior del cursos (como en Windows). Usando **history** podemos ver la lista de comandos. Con el signo de admiración y el número de orden invocaremos directamente al comando deseado



Hay otros comandos como **sort**, que nos pueden se muy útiles. Vemos en el ejemplo como ordenar el fichero de contraseñas solo por el primer parámetro identificando el separador



El comando **awk** es un comando curioso, que nos permite extraer una columna concreta de un fichero (puede ser la entrada que le llega por una tubería)

awk '{ print $2 }' < fichero

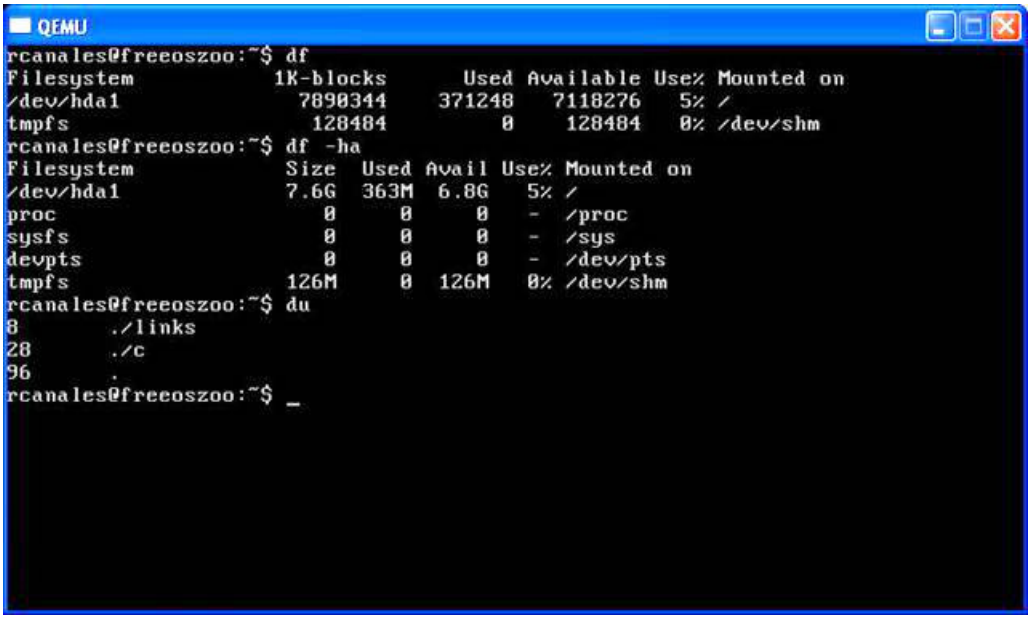
ls -lF | grep m | awk '{ print $2 }' < fichero

Si queremos ver los nombres de los usuarios datos de alta en el sistema, podríamos ejecutar un comando como este (indicando el carácter separador):

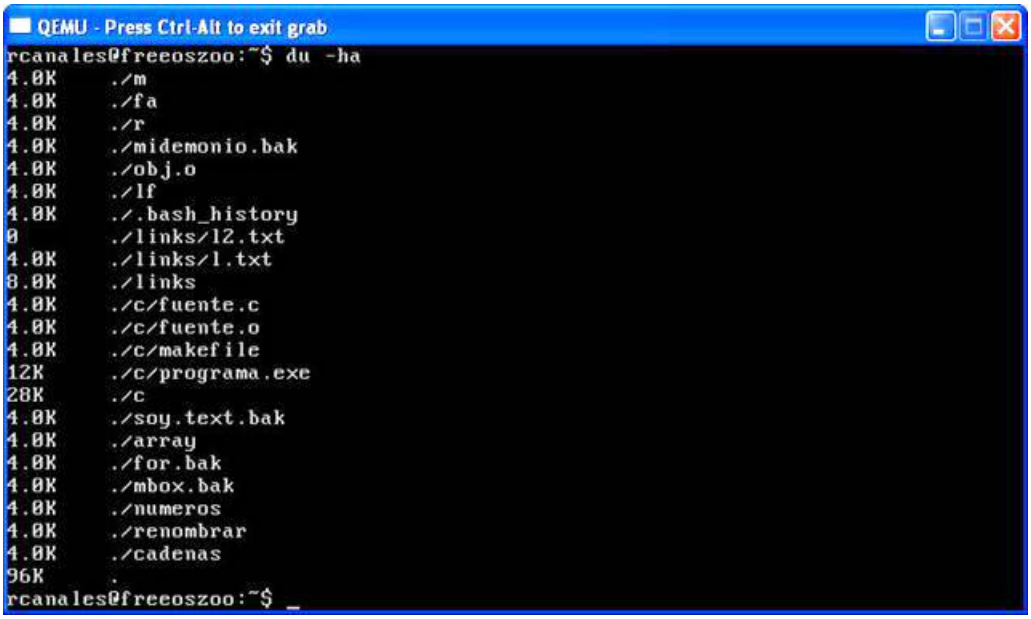
cat passwd | awk -F':' '{printf $1}'

Comando de Disco

Para ver los discos montados podemos usar el comando **df** y para que sea legible por humanos, usaremos el comando **-fu**



Si queremos saber el espacio usado en disco por distintos ficheros/directorio usaremos **du -ha**



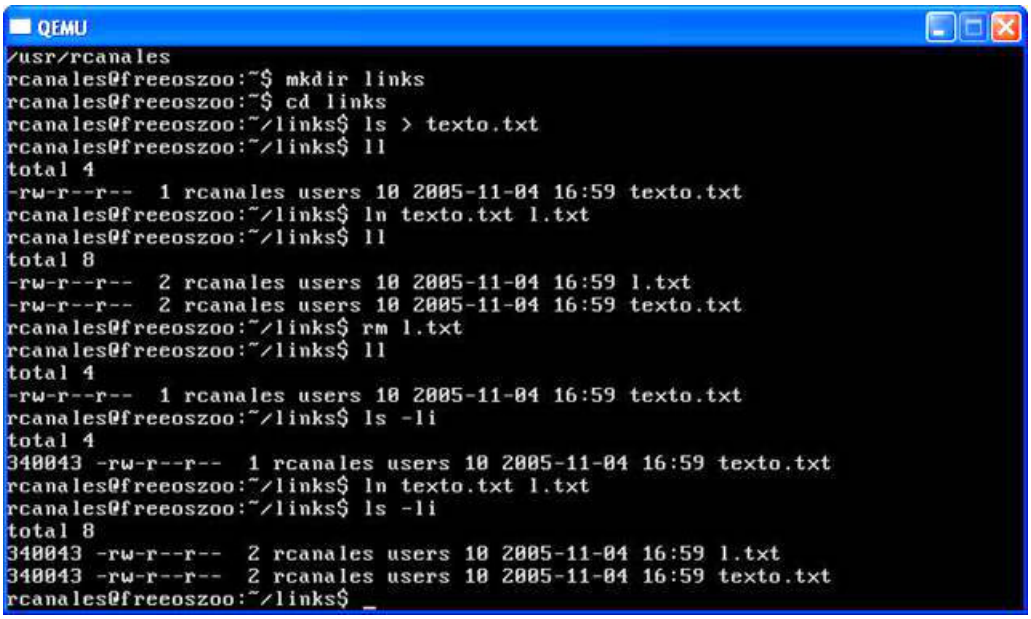
Links a ficheros

Podemos establecer links a fichero y/o directorios (solo débiles). Hay dos tipos de links, los fuertes y los débiles.

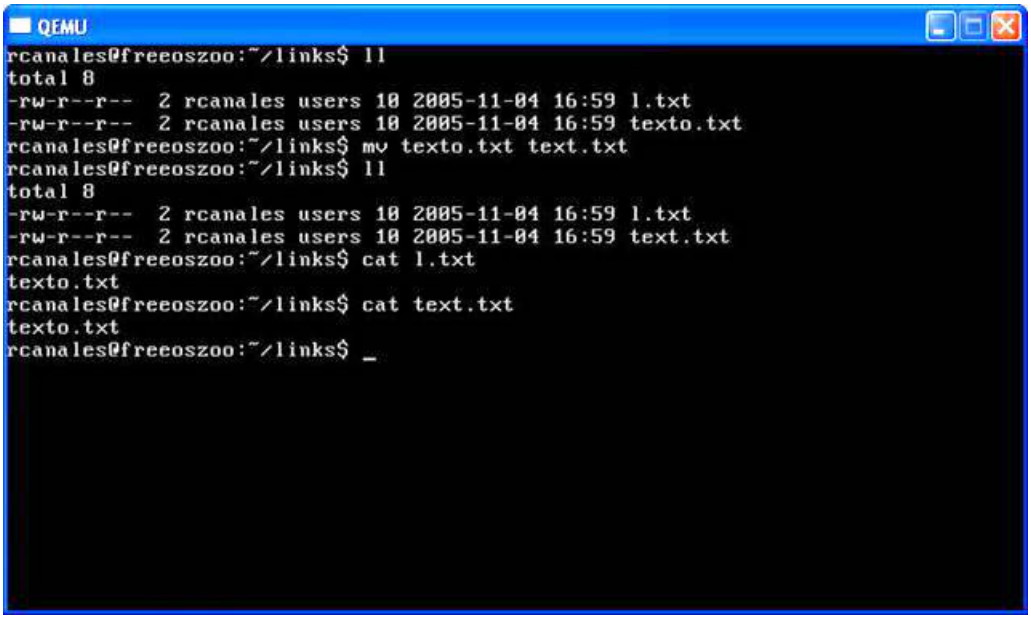
En los links fuertes, enganchamos con el identificador interno del elemento apuntado (identificador de nodo). Esto significa que aunque se cambie el nombre de recurso ligado, el link sigue siendo válido (evito ser más técnico aposta) .

Los links fuertes se hacen con el comando **ln nodoexistente nombredellink** (no se puede hacer entre distintos sistemas de ficheros)

En la siguiente captura solo se trata de representar que cuando realizamos un **ls** podemos ver que aumenta el contador de links al nodo enlazado (obviamente se borrara el fichero cuando demos la orden y el contador sea cero)

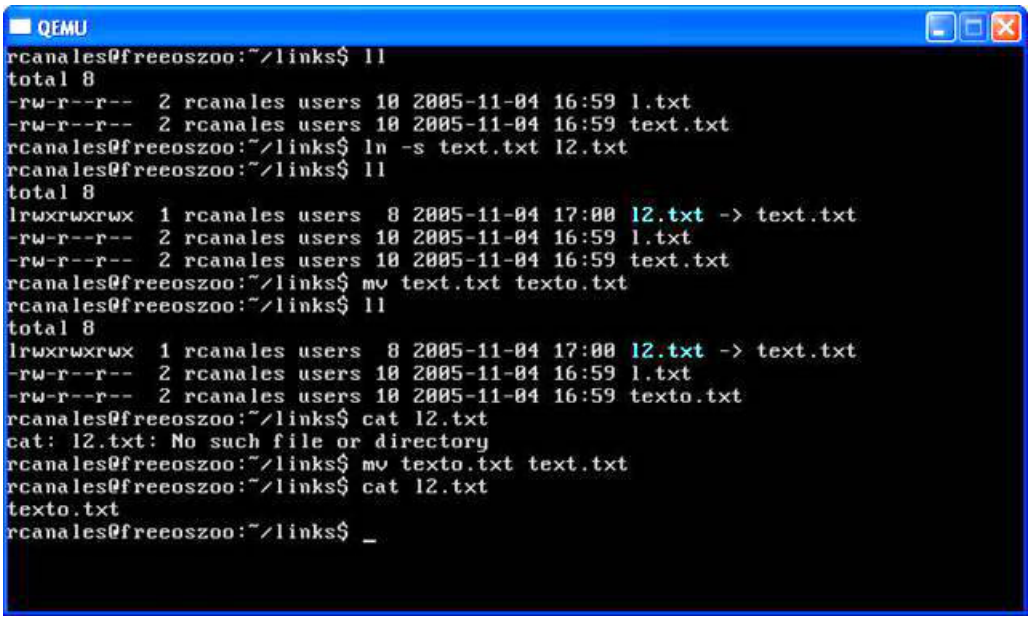


Si cambiamos de nombre el fichero original, no pasa nada porque realmente se está enlazando al nodo interno

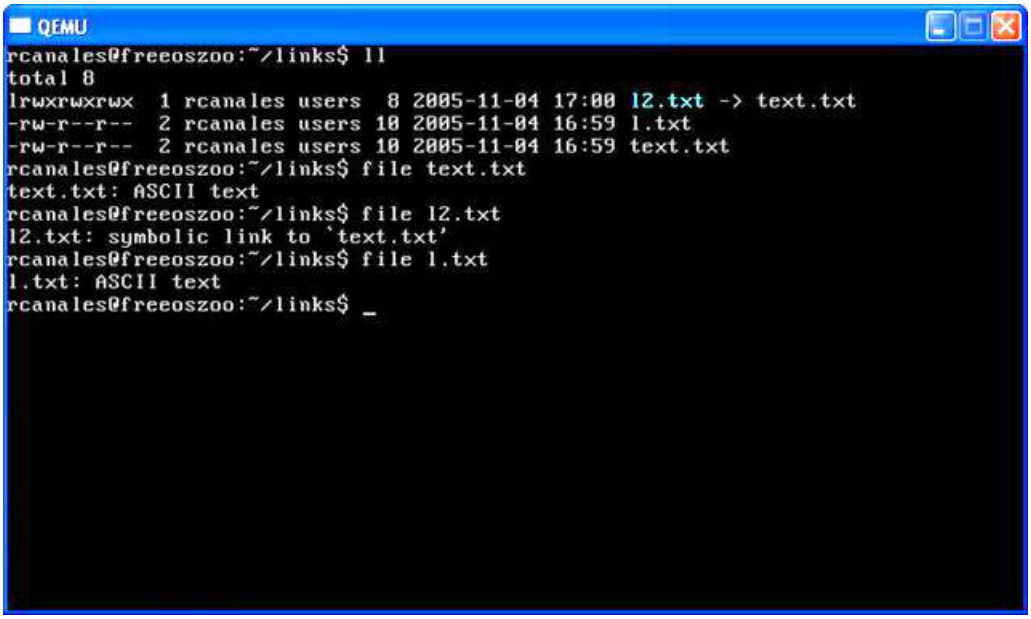


Los enlaces débiles vienen a ser como los accesos directos en Windows, enlazamos a un nombre. si cambiamos el nombre del fichero enlazado, perdemos el link (además esto no incrementa el contador de referencia a un nodo).

Para hacer links débiles se usa el comando **ln -s nodoexistente nombredellink**



Si reestablecemos el nombre del fichero, el link vuelve a funcionar



Copiar y mover directorios y ficheros

Para el que trabaje en entornos Windows, el sistema de ficheros de Linux puede ser un poco complejo.

Hay que diferenciar que en Windows tenemos distintas árboles (uno por cada unidad física o lógica). En Linux tenemos un solo árbol sobre el que montamos los distintos discos.

Si queremos copiar y/o mover los datos de un disco físico a otro ... no es tan sencillo como en Windows (estos son temas avanzados).

Os invito a que investiguéis en este aspecto ... aunque os dejo un par de comando a investigar

cp -a /directorio/origen /directorio/destino (en caso de no existir enlaces)

find . -depth -print0 | afio -p -xv -0a (en caso de existir links)

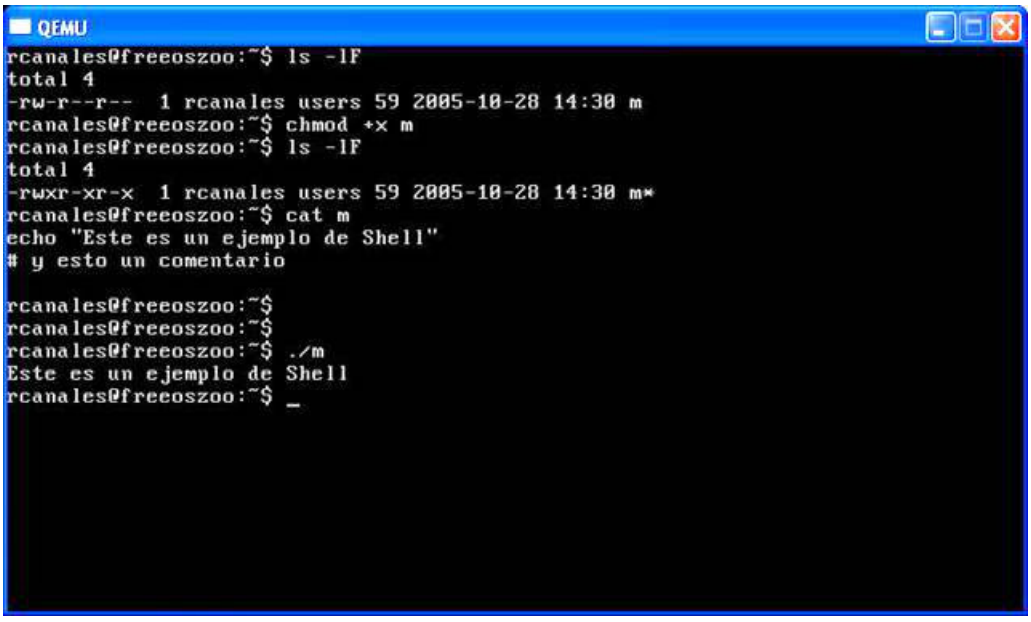
También investigar el comando **cpio**

Programación con shell-script

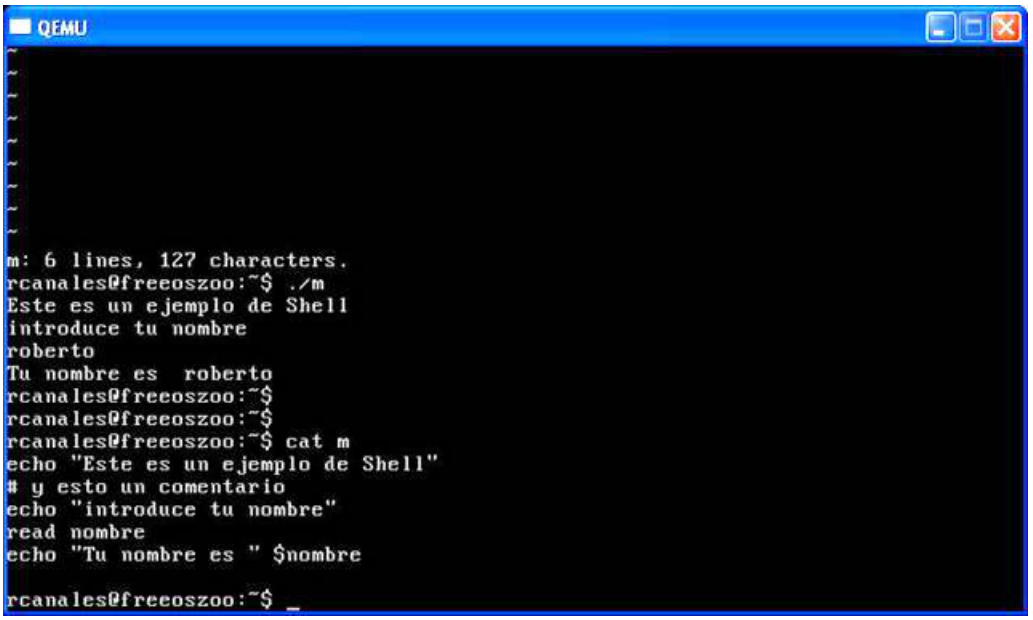
Una de las cosas más agradecidas que tiene Unix, es que con el Shell se pueden hacer verdaderos programas avanzados con unos conocimientos relativamente básicos.

Con el editor **vi** (o vim en la actualidad) podemos escribir un pequeño script.

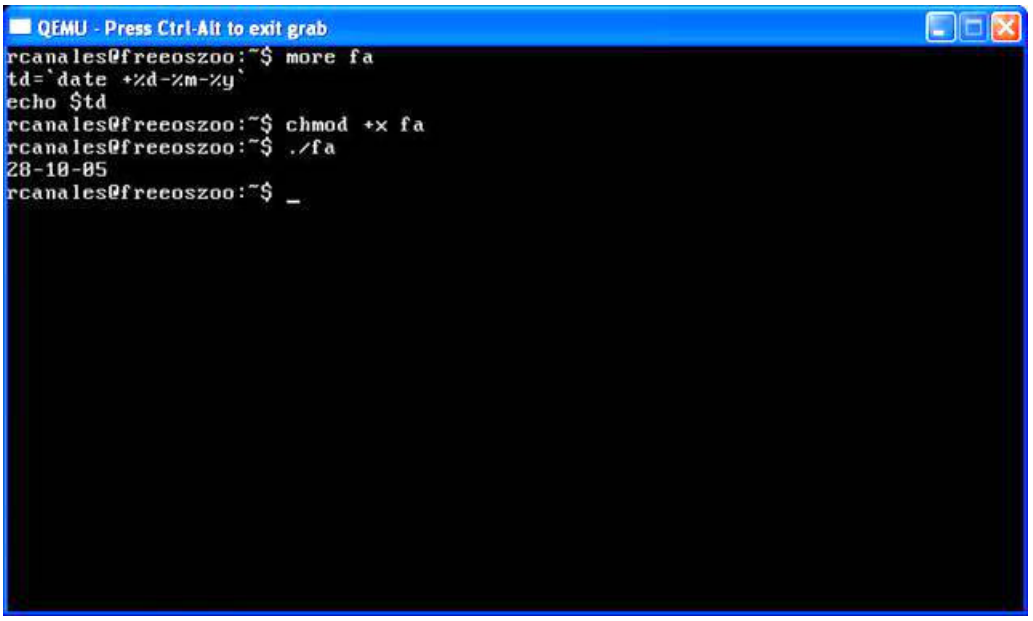
Con **chmod +x** podemos asignar privilegios de ejecución al fichero. Os recomiendo que uséis **ls -F** para ver fácilmente los ficheros ejecutables



Los comentarios en los Shell-scritps se hacen con #



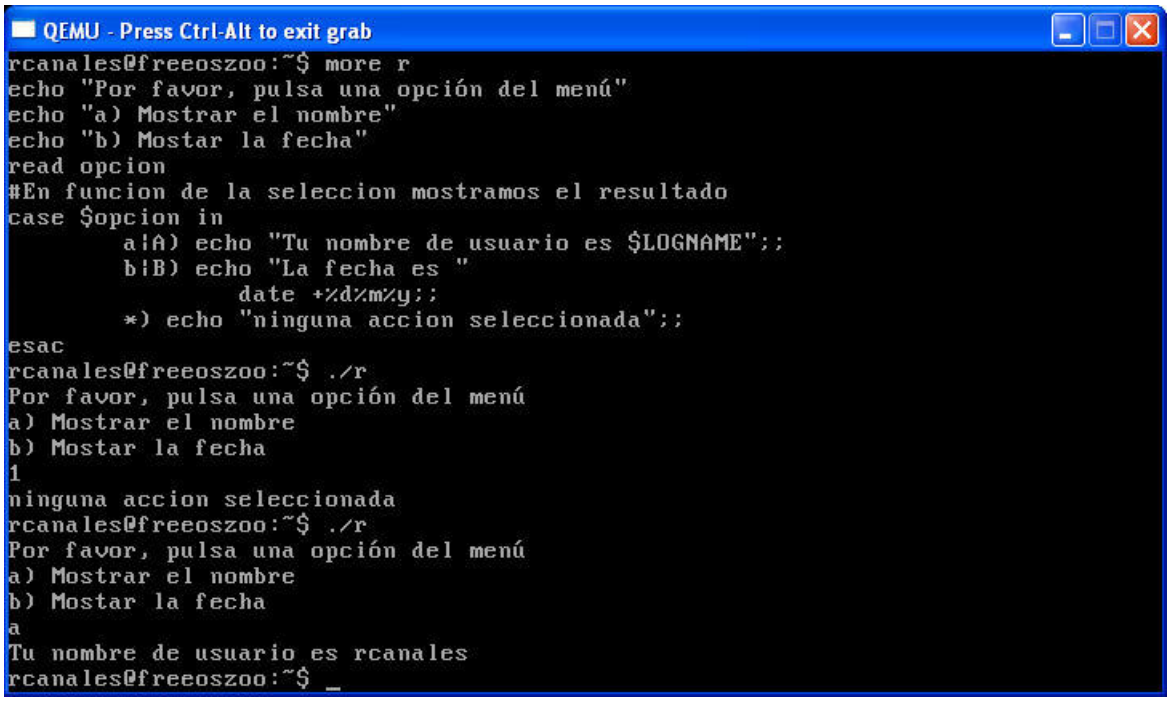
Dentro de un Scritp, la tilde invertida nos sirve para indicar que queremos obtener el resultado de la ejecución de un comando.



Este es el conjunto de variables disponibles dentro de nuestros scripts:

- $# - Cantidad de parámetros que el script recibe.
  $* - Todos los parámetros recibidos separados por $IFS (espacio por defecto).
  $- Opciones que se pasaron al script durante su ejecución.
  $$ - Identificación del proceso del intérprete de comandos.
  $? - Código de retorno del último programa ejecutado
  $! - Identificación del proceso del último comando que se ejecutó en segundo plano.
  $0 - Nombre del script.

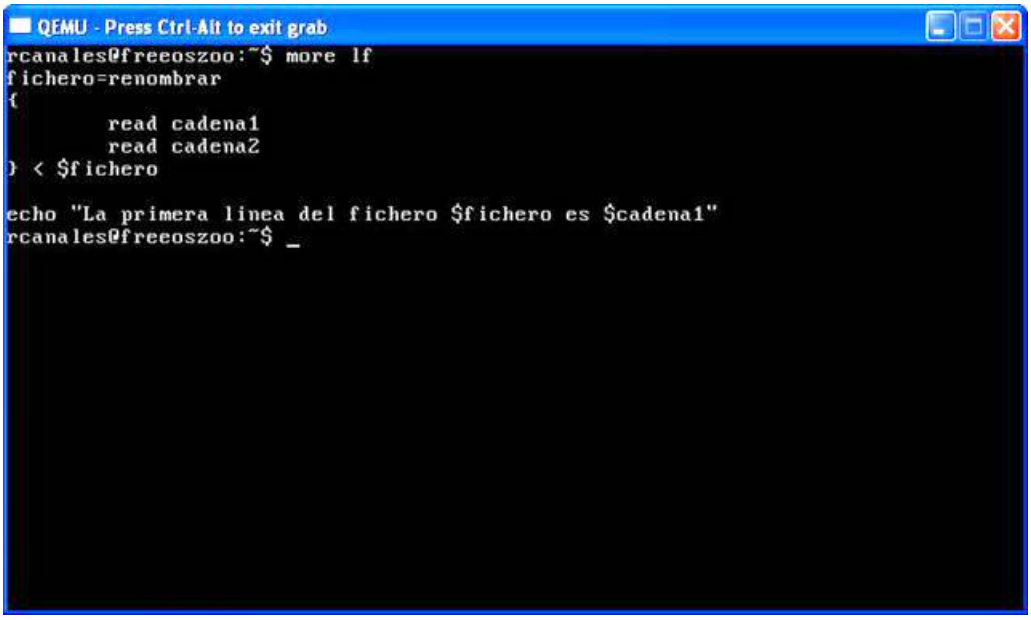
Hacer un menú con un Shell es trivial con las instrucciones **case** ..



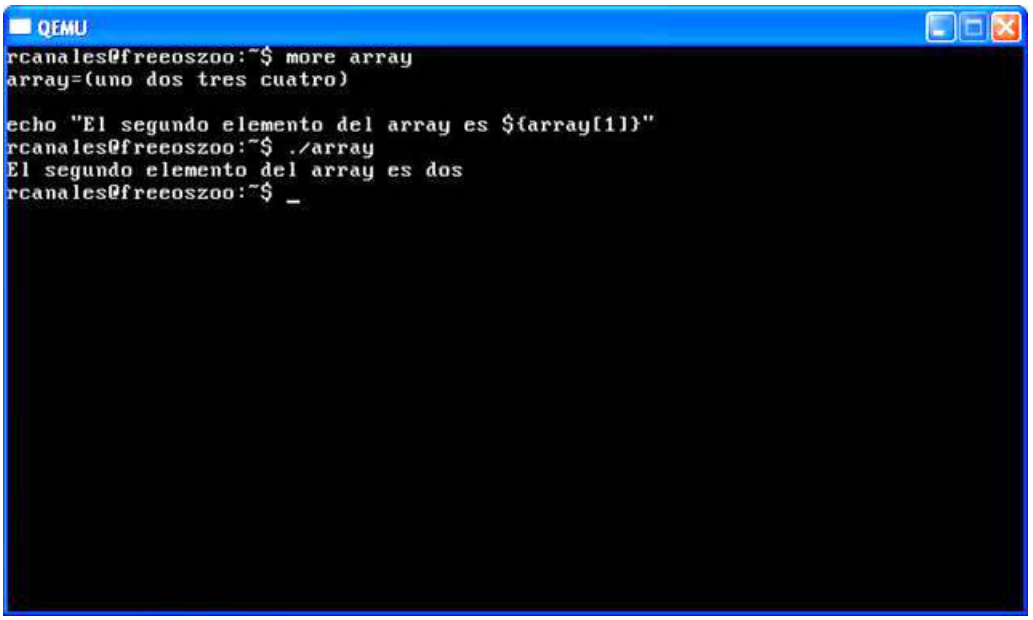
Como en cualquier lenguaje disponemos de un if y podemos hacer comprobaciones con las opciones:

-b fichero = Comprueba si fichero es un dispositivo orientado a bloques
-c fichero = Comprueba si fichero es un dispositivo orientado a caracteres
-d fichero = Comprueba si fichero es un directorio
-f fichero = Comprueba si fichero es un fichero normal
-r fichero = Comprueba si fichero es legible por el proceso
-w fichero = Comprueba si fichero es escribible por el proceso
-x fichero = Comprueba si fichero es ejecutable

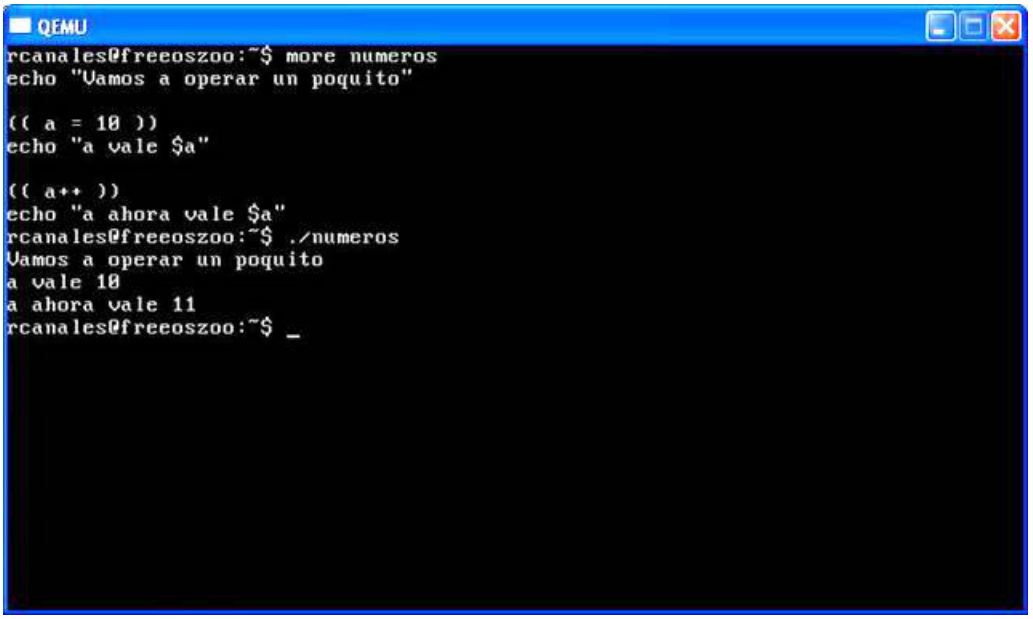
Con el comando **read** podemos leer lo que el usuario nos escriba. Podemos usar las redirecciones para procesar el contenido de un fichero.



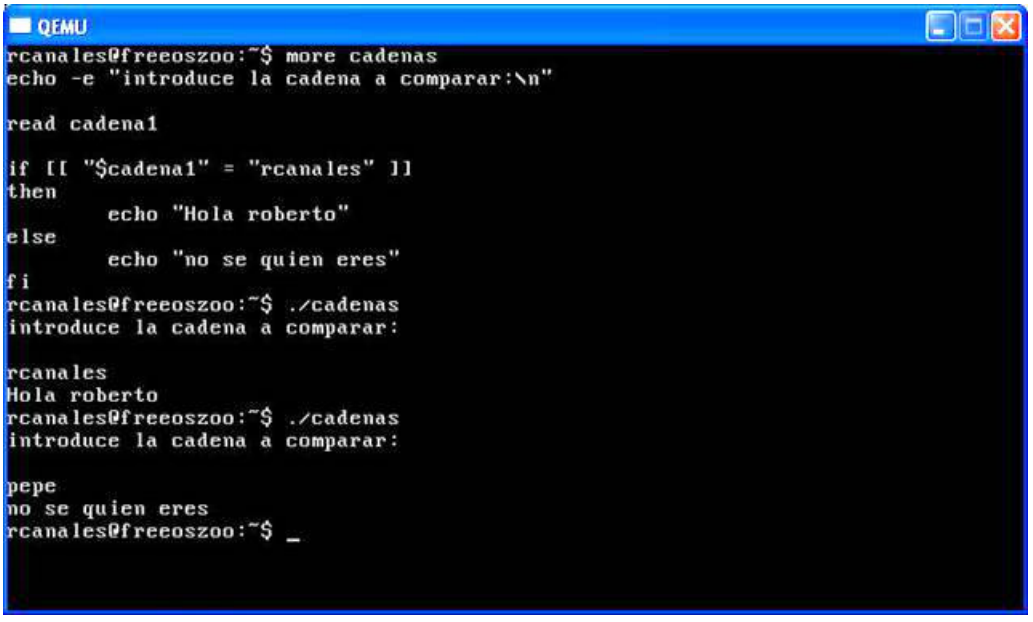
Como en casi todos los lenguajes, disponemos de arrays ..



Podemos inicializar variables y operar con ellas de un modo numérico con los dobles paréntesis (si le echamos un poco de imaginación, veremos los bucles **for** desde otra perspectiva en los **shell** de **Unix**)

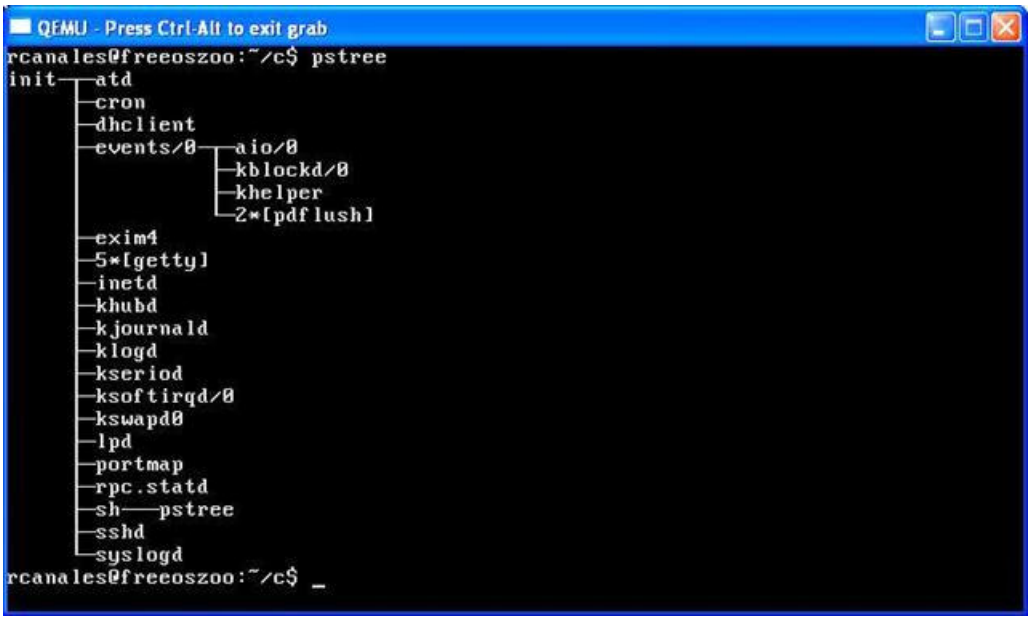


Con **echo -e** activamos el procesamiento de las secuencias de escape como **\n** para una nueva línea

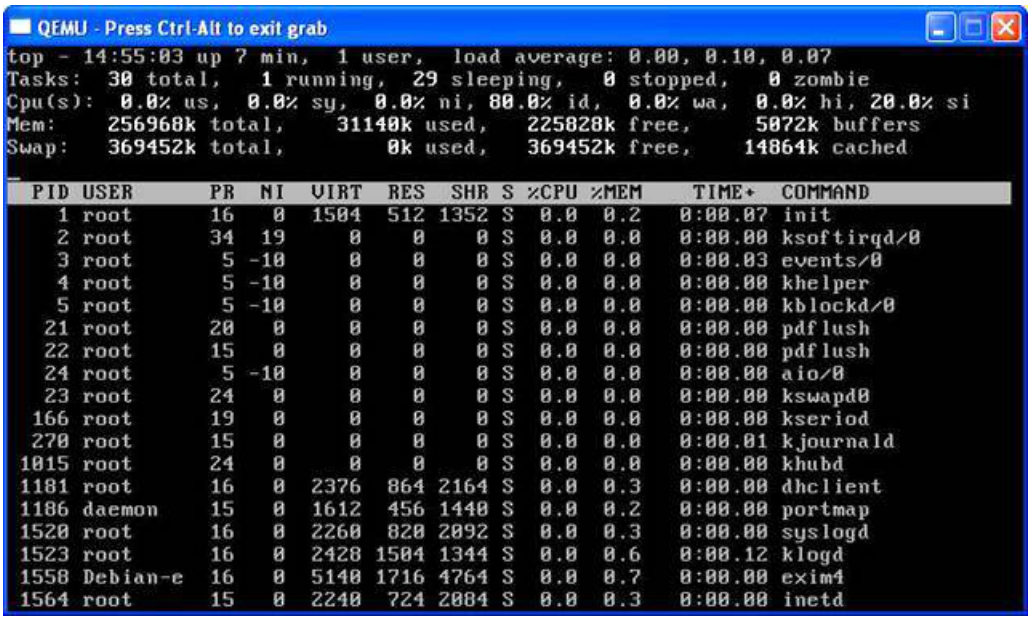


Procesos

Con el comando **ps** podemos ver los procesos activos. Podemos ver el árbol de procesos activos con pstree.



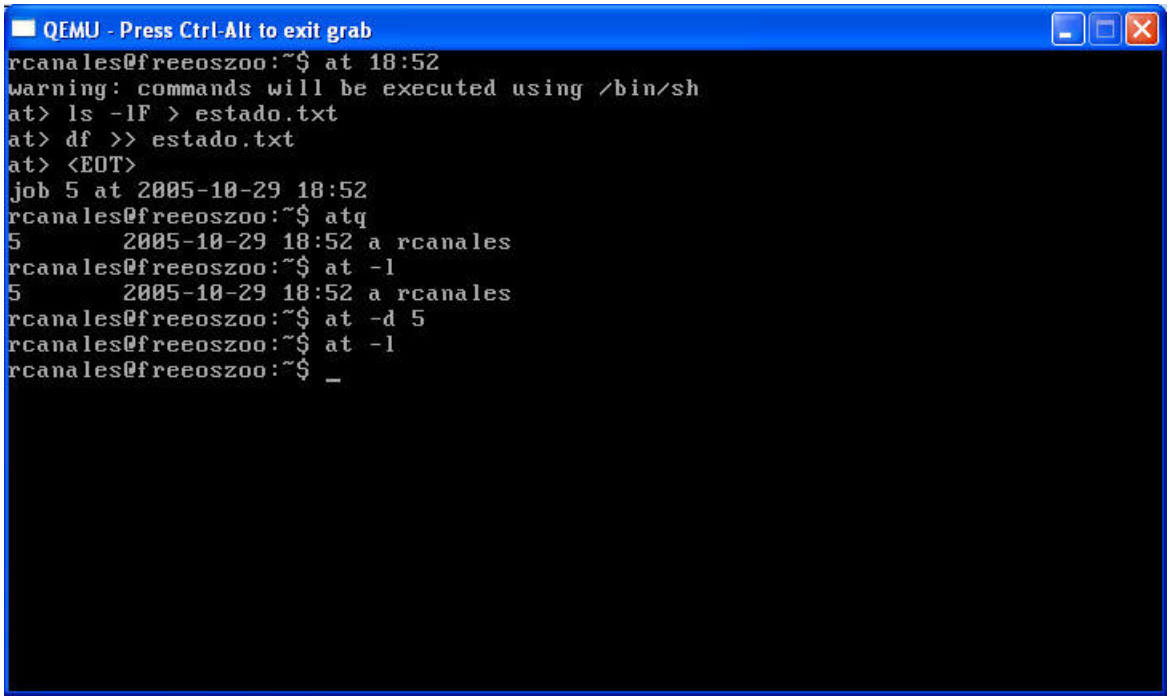
Ver estado de la memoria con el comando **top**



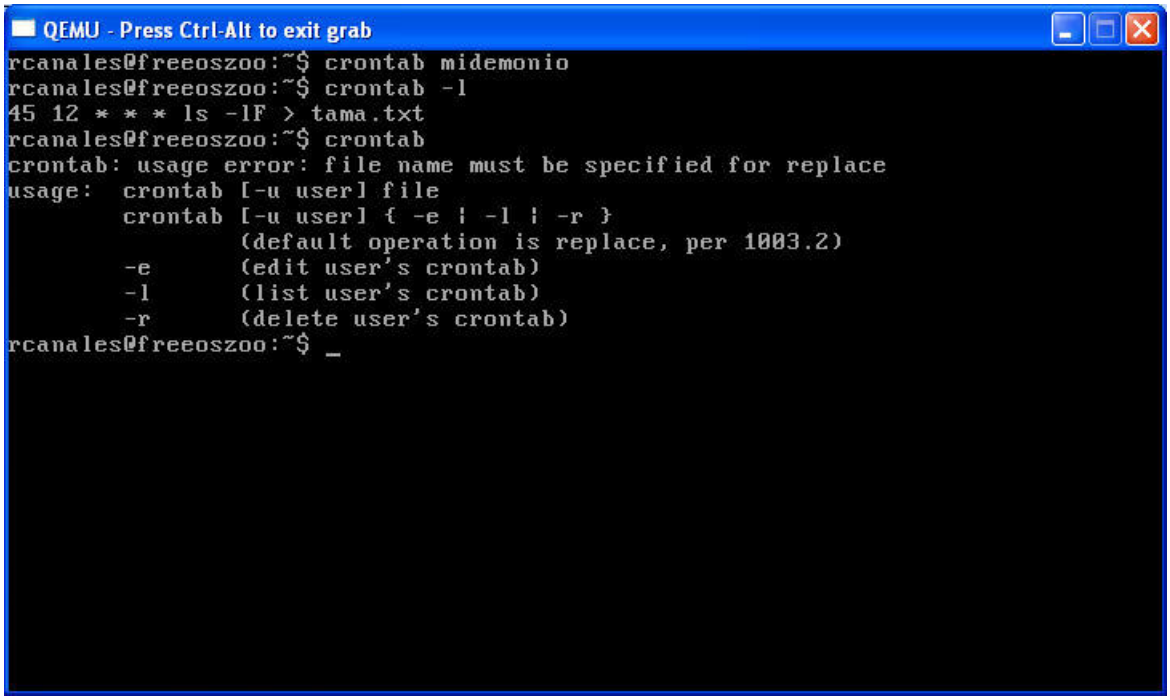
Programación de tareas en Linux

Un comando o conjunto de comando se puede ejecutar de un modo inmediato o de un modo diferido.

Con el comando **at**, podemos definir cuando queremos que se ejecute un comando



Si queremos que se ejecuten comandos de un modo periódico, podemos usar el **cron**, a través del comando **crontab**

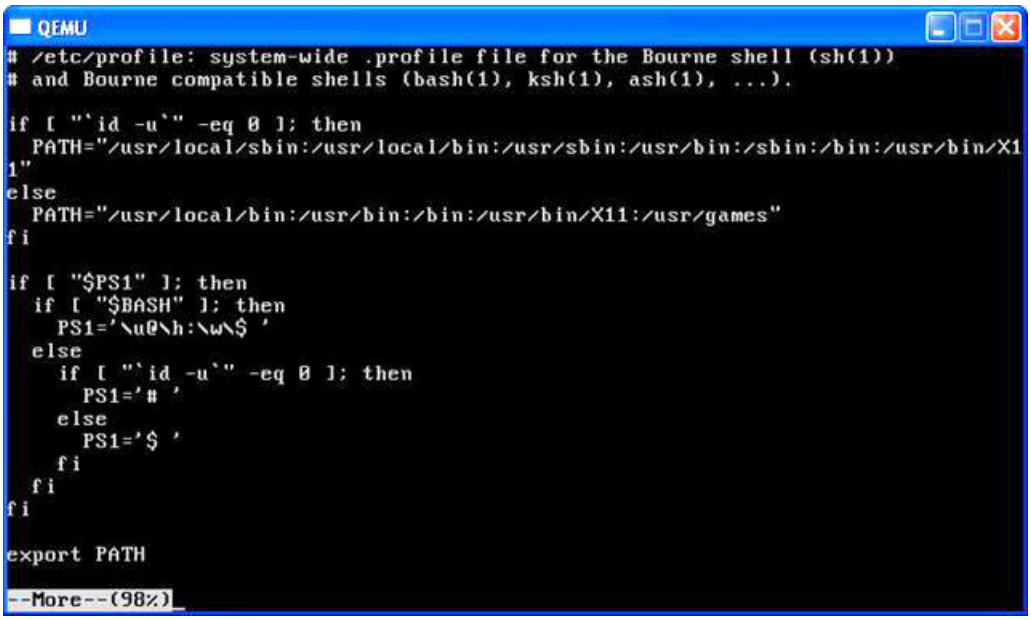


No olvidar que es muy posible que no se ejecuten los comandos si se han configurados opciones de seguridad (el usuario debe estar en el fichero /etc/crontab.r/cron.allow, si este existe)

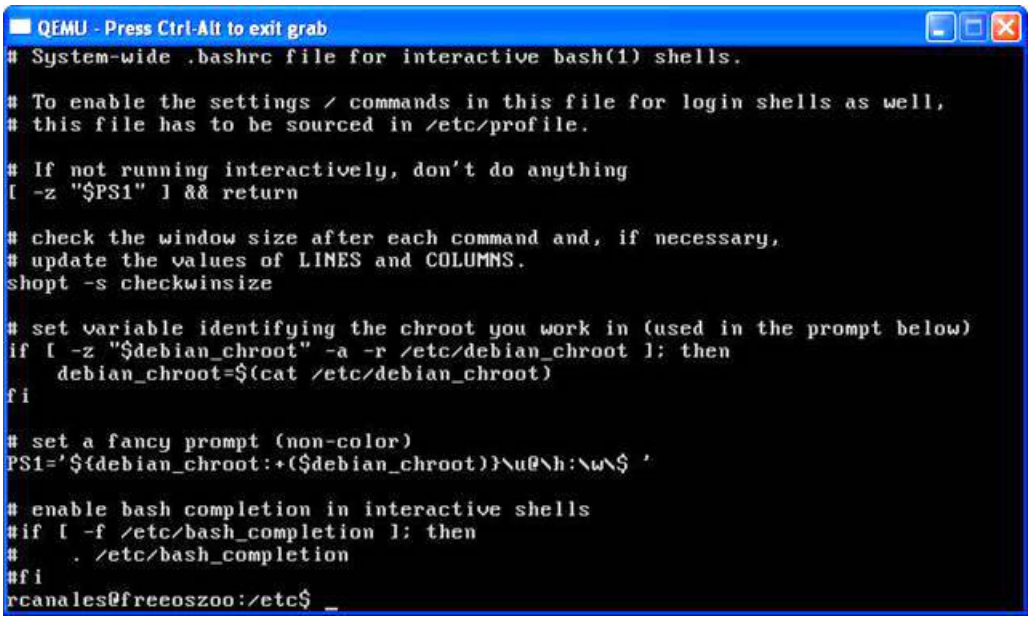
Otras configuraciones

¿De donde sale la variable PATH en Linux?

more /etc/profile



Definición del prompt **/etc/bash.bashrc**



Salir ordenadamente de Linux

```
# shutdown -h now

# poweroff -i -f
```

Conclusiones

En Internet hay miles de manuales y tutoriales ... os indico algunos de ellos para que sigáis investigando...

Links Interesantes

http://www.europe.redhat.com/documentation/rhl6.2/gsg-es/index.php3

http://www.sinuh.org/content/view/94/27/

http://www.adictosaltrabajo.com/indexg.php?noimages=SI&pagina=listacat&id=20 (en www.adictosaltrabajo.com)

http://www.wernitz.net/docs_debian/

http://www.tldp.org/LDP/abs/html/

Si desea contratar formación, consultoria o desarrollo de piezas a medida puede contactar con

Creatividad Internet

Autentia S.L. Somos expertos en:
J2EE, C++, OOP, UML, Vignette, Creatividad ..
y muchas otras cosas

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
e-mail	
	Enviar

Otros Tutoriales Recomendados (También ver todos)

Nombre Corto	Descripción
Compartir impresoras y ficheros con Linux	Cesar Crespo Martín y Alejandro Perez García nos enseñan como realizar la compartición de impresoras y ficheros con Linux, CUPS y SAMBA con clientes Windows.
Detección de errores Java con FindBugs	Os mostramos como instalar y utilizar FindBugs, una excelente herramienta para analizar, de un modo estático, posibles problemas en vuestro código Java
Nombrar dispositivos USB en Debian	En este tutorial vamos a ver como podemos asignar nombres a los dispositivos USB en Debian, independientemente del nombre que le de el kernel.
Gestión de Errores con Bugzilla	Alejandro Pérez nos enseña como instalar (en Debian) y utilizar Bugzilla, una herramienta gratuita de gestión de errores.
Uso de Tiles en Struts	Os mostramos como utilizar el sistema de plantillas proporcionado por Struts (tiles)
Fuentes Gnome (GTK) con KDE (Qt)	Alejandro Pérez nos enseña como configurar las fuentes de aplicaciones Gnome (GTK) si usamos KDE (Qt)
Informar a Debian de la instalación de Java	En este tutorial os mostramos como informar a Debian GNU/Linux de nuestra Máquina Virtual Java, no instalada desde paquetes Debian.
Gestión de contenidos y errores comunes	Os explicamos en que consiste la gestión de contenidos y cuales son los errores cometidos por multitud de empresas a la hora de abordar su implantación
Instalación LAMP en Debian	Alejandro Perez nos enseña como instalar nuestro entorno Apache, MySQL y php en Debian
Herramientas Gratuitas UML	Os mostramos como obtener algunas herramientas gratuitas UML, ArgoUML y Poseidon.

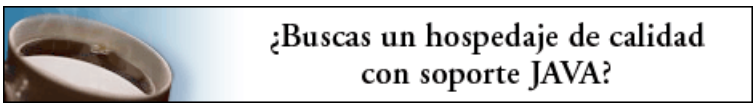
Nota: Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento.

Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores.

En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo.

Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.

Patrocinados por enredados.com .... Hosting en Castellano con soporte Java/J2EE



www.AdictosAlTrabajo.com Opimizado 800X600