

ÍNDICE DE CONTENIDO

Tema 6. Estados de Ejecución y Secuencias de Arranque.....	3
6.1 El Proceso de arranque.....	4
6.2 El Proceso INIT.....	4
6.2.1 SystemV.....	4
6.2.1.1 Niveles de Ejecucion.....	5
6.2.2 BSD.....	7
6.3 El fichero init.d.....	8
6.4 El fichero rcN.d.....	9
6.5 El fichero inittab.....	10
6.6 El fichero rc.sysinit.....	12
6.7 El fichero rc.local.....	12
6.8 Niveles de Ejecucion.....	13
6.8.1 Nivel 0 –Parada Del Sistema.....	13
6.8.2 Nivel 1 o S -Monousuario o Single User.....	13
6.8.3 Nivel 2 -Multiusuario sin Red.....	14
6.8.4 Nivel 3 -Multiusuario con Red.....	14
6.8.5 Nivel 4. -Sin Uso.....	14
6.8.6 Nivel 5. -Multiusuario Grafico.....	15
6.8.7 Nivel 6. -Reinicio del Sistema.....	15
6.9 Comando chkconfig.....	16
6.10 Levantando, deteniendo y reiniciando servicios.....	16

Información de Derechos reservados de esta publicación.

Reconocimiento-NoComercial-CompartirIgual 2.1

Usted es libre de:

- Copiar, Distribuir y Comunicar públicamente la obra

Bajo las condiciones siguientes:



Reconocimiento. Debe reconocer y citar al autor original.



No comercial. No puede utilizar esta obra para fines comerciales.



Compartir bajo la misma licencia. Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta.

- Al reutilizar o distribuir la obra, tiene que dejar bien claro los términos de la licencia de esta obra.
- alguna de estas condiciones puede no aplicarse si se obtiene el permiso del titular de los derechos de autor

Los derechos derivados de usos legítimos u otras limitaciones no se ven afectados por lo anterior.

Reconocimiento-NoComercial-CompartirIgual 2.1

Tema 6. Estados de Ejecución y Secuencias de Arranque

6.1 El Proceso de arranque

El proceso de arranque de un sistema operativo Linux se inicializa de la siguiente manera:

Cuando usted enciende su servidor o su computadora personal, esta hace que el BIOS de su equipo inicie las operaciones relacionadas con el arranque. El **BIOS** (Basic Input Output System) es un pequeño programa escrito en lenguaje ensamblador cuya función es cargar el sistema operativo en la memoria RAM (Random Access Memory), una vez que el BIOS carga el sistema operativo en RAM este inicia un proceso llamado **POST** (Power On Self Test) el cual es un proceso de diagnostico y verificación de los componentes de entrada y salida de un servidor o computadora y se encarga de configurar y diagnosticar el estado del hardware, una vez verificado el hardware se inicia la fase de arranque del sistema (**bootstrapping**) el cual cede el control al **GRUB** (Grand Unified Bootloader), el GRUB es un gestor de arranque que hace uso de un menú gráfico que permite elegir el Sistema Operativo que se desea arrancar; Así mismo, el GRUB realiza las siguientes tareas:

1. Cargar el kernel en memoria.
2. Cargar el sistema de ficheros virtual **initrd** el cual es usado tipicamente para hacer los arreglos necesarios antes de que el sistema de ficheros raíz pueda ser montado
3. Pasarle los argumentos **runlevel** e **init** al kernel
4. Comenzar la ejecución del **kernel**

Al terminar de ejecutar todas las tareas anteriores el GRUB le cede el control total del arranque al kernel y este a su vez se encarga de realizar la llamada a la función **starup** la cual tiene como función detectar el tipo de CPU con el que el equipo cuenta así como de lo principal del sistema operativo, como el manejo de memoria, planificador de tareas, entradas y salidas, comunicación interprocesos, y demás sistemas de control, a partir de este momento se ejecuta el proceso **INIT**.

6.2 El Proceso INIT

INIT es el primer proceso en ejecutarse despues de la carga del kernel de linux e implementa dos modelos bajo los cuales puede trabajar, estos son

1. SystemV
2. BSD

Estos modelos son arrancados por un programa (script) de arranque que establece como deben inicializarse los diferentes servicios, programas o registros que sean necesarios para que el sistema funcione como el administrador lo requiere.

Explicaremos brevemente como es que trabajan estos modelos

6.2.1 SystemV

Es un modelo usado para controlar el inicio y apagado del sistema y fue originalmente desarrollado por la compañía estadounidense de telecomunicaciones AT&T.

SystemV fue una de las versiones del sistema operativo Unix que se encargaba de controlar el arranque de los programas en el instante de inicio del equipo. Este modelo es considerado por muchos como facil, potente y flexible en comparacion con el sistema de inicio **BSD**

Existen cuatro versiones release de SystemV (**SVR**) , las cuales son:

1. SVR1.-Primera version de SystemV lanzada en 1984, incluía el editor de textos Vi
2. SVR2.-Incluye mejoras con respecto al nucleo el cual esta implementado como memoria virtual paginada, el sistema operativo Apple esta basado en este modelo.
3. SVR3.-Incluye mejoras en el sistema de ficheros asi como una nueva API de red, el sistema operativo AIX de IBM hace uso de este modelo
4. SVR4.- Fue la versión más popular de SVR asi como la fuente de varias características comunes del sistema operativo Unix, como el script **/etc/init.d**

6.2.1.1 Niveles de Ejecucion

Los niveles de ejecucion en SystemV describen ciertos estados del equipo los cuales se caracterizan por ejecutar ciertos procesos. En general existen 8 niveles de ejecucion los cuales van del 0 al 6 y S o s, que son alias del mismo nivel de ejecucion, de estos ocho niveles, tres son considerados reservados, estos son:

0.- Halt

1.-Single user mode

6.-Reboot

Aparte de los niveles de ejecucion 0,1 y 6 todos los sistemas operativos Linux tratan a los niveles de ejecucion un poco diferente. El denominador comun de todas las distribuciones linux es el fichero

/etc/inittab

el cual define lo que hace cada nivel de ejecucion.

A continuacion un ejemplo de cuantos niveles de ejecucion tienen cada una de las distribuciones mas importantes de linux, asi como del sistema operativo solaris y AIX

Sistema Operativo	Niveles de ejecucion por default
	2
	2
	3

Sistema Operativo	Niveles de ejecucion por default
	5
	3 o 5
	3 o 5
	3
	5
	2
	2

En la mayoría de los sistemas operativos linux los usuarios pueden saber bajo que nivel de ejecucion estan trabajando tecleando en una consola y como root lo siguiente:

```
[root@localhost]$ runlevel
N 5
```

Existen tambien los ficheros llamados **rcN.d** en donde la letra **N** representa cada uno de los niveles de ejecucion en los que trabaja **init.d**, la funcion de estos ficheros se explicara mas a detalle en el siguiente tema.

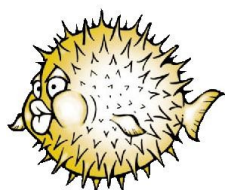
6.2.2 BSD

El modelo BSD init se ejecuta mediante el script de inicializacion situado en la ruta

```
/etc/rc
```

Algunos de los sistemas operativos que se basan en este modelo son los basados en BSD como:

- FreeBSD
- NetBSD
- OpenBSD
- DragonFlyBSD
- DesktopBSD
- PCBSD



OpenBSD



DesktopBSD



DragonFlyBSD

PCBSD

6.3 El fichero init.d

En este fichero se encuentran todos los scripts encargados de levantar cada uno de los servicios del servidor.

La ubicación de este fichero esta localizada en:



Algunos de los servicios que podemos encontrar en el fichero init.d son los referentes a:

- Servidor Web Apache ---->httpd
- Servidor Samba ----->smb
- Servidor de Correo --->sendmail
- Servidor DHCP ---->dhcpd
- Servidor DNS ---->named
- Manejador de Base de Datos MySQL ---->mysqld

```
[root@localhost ~]# cd /etc/rc.d/init.d/
[root@localhost init.d]# ls -l
total 428
-rwxr-xr-x 1 root root 2974 jun 23 10:18 dhcpd
.
.
.
-rwxr-xr-x 1 root root 3099 feb 25 2008 httpd
-rwxr-xr-x 1 root root 4239 mar 3 2008 mysqld
.
.
.
-rwxr-xr-- 1 root root 6154 ago 6 05:05 named
-rwxr-xr-x 1 root root 1745 sep 18 10:26 smb
-rwxr-xr-x 1 root root 4112 mar 29 2008 sendmail
```


6.4 El fichero rcN.d

rcN.d es un conjunto de directorios que representan cada uno de los niveles de ejecución del sistema operativo. Estos directorios a su vez contienen un conjunto de enlaces simbólicos a los scripts del directorio **/etc/rc.d/init.d**

La función que desempeñan estos directorios es organizar la manera en como los servicios de un servidor son levantados, como por ejemplo, imaginemos que tenemos instalado un servidor Web apache, y que lo tenemos configurado para que trabaje en los niveles de ejecución 3 y 5, por ende deberíamos poder observar dichos enlaces simbólicos en las rutas

- /etc/rc.d/rc3.d
- /etc/rc.d/rc5.d

Otra de las características de estos enlaces simbólicos es la sintaxis de sus propiedades. Esta sintaxis está conformada por 3 parámetros

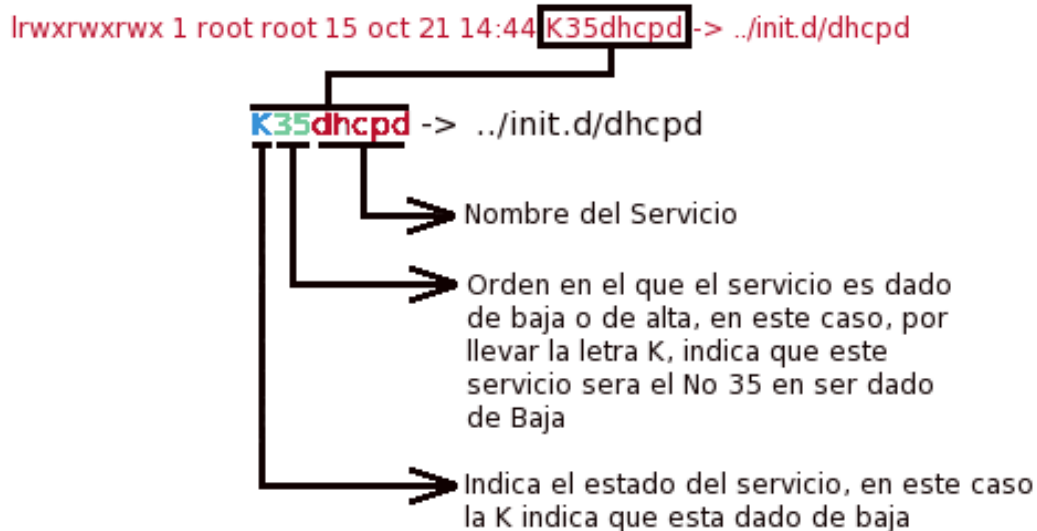
1. El estado del servicio, los cuales son representados con dos variables:
 - La letra K.-Esta letra representa que el servicio está dado de baja
 - La letra S.-Esta letra representa que el servicio está dado de alta
2. El orden en el que es arrancado el servicio.- Este parámetro indica el orden en el que los servicios deben ser dados de alta o de baja
3. El nombre del servicio

Un ejemplo de esto lo podemos observar de la siguiente tabla, la cual la tomamos de la ruta **/etc/rc.d/rc5.d**, lo cual indica que los scripts dentro de esta carpeta se ejecutan en el nivel de ejecución 5

```

reaperE@localhost:/etc/rc5.d
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
[reaperE@localhost rc5.d]$ ls -l
total 0
lrwxrwxrwx 1 root root 16 oct 1 04:54 K01smartd -> ../init.d/smartd
lrwxrwxrwx 1 root root 19 oct 1 04:54 K05sasauthd -> ../init.d/saslauthd
lrwxrwxrwx 1 root root 19 oct 1 04:50 K10dc_server -> ../init.d/dc_server
lrwxrwxrwx 1 root root 16 oct 1 04:50 K10psacct -> ../init.d/psacct
lrwxrwxrwx 1 root root 19 oct 1 04:50 K12dc_client -> ../init.d/dc_client
lrwxrwxrwx 1 root root 13 oct 1 04:50 K15gpm -> ../init.d/gpm
lrwxrwxrwx 1 root root 15 oct 1 04:54 K15httpd -> ../init.d/httpd
lrwxrwxrwx 1 root root 13 oct 1 04:56 K20nfs -> ../init.d/nfs
lrwxrwxrwx 1 root root 14 oct 1 04:56 K24irda -> ../init.d/irda
lrwxrwxrwx 1 root root 15 oct 1 05:05 K25squid -> ../init.d/squid
lrwxrwxrwx 1 root root 15 oct 21 14:44 K35dhcpd -> ../init.d/dhcpd
lrwxrwxrwx 1 root root 18 oct 21 14:44 K35dhcrelay -> ../init.d/dhcrelay
  
```

Los detalles del renglón subrayado se explican a continuación:



6.5 El fichero inittab

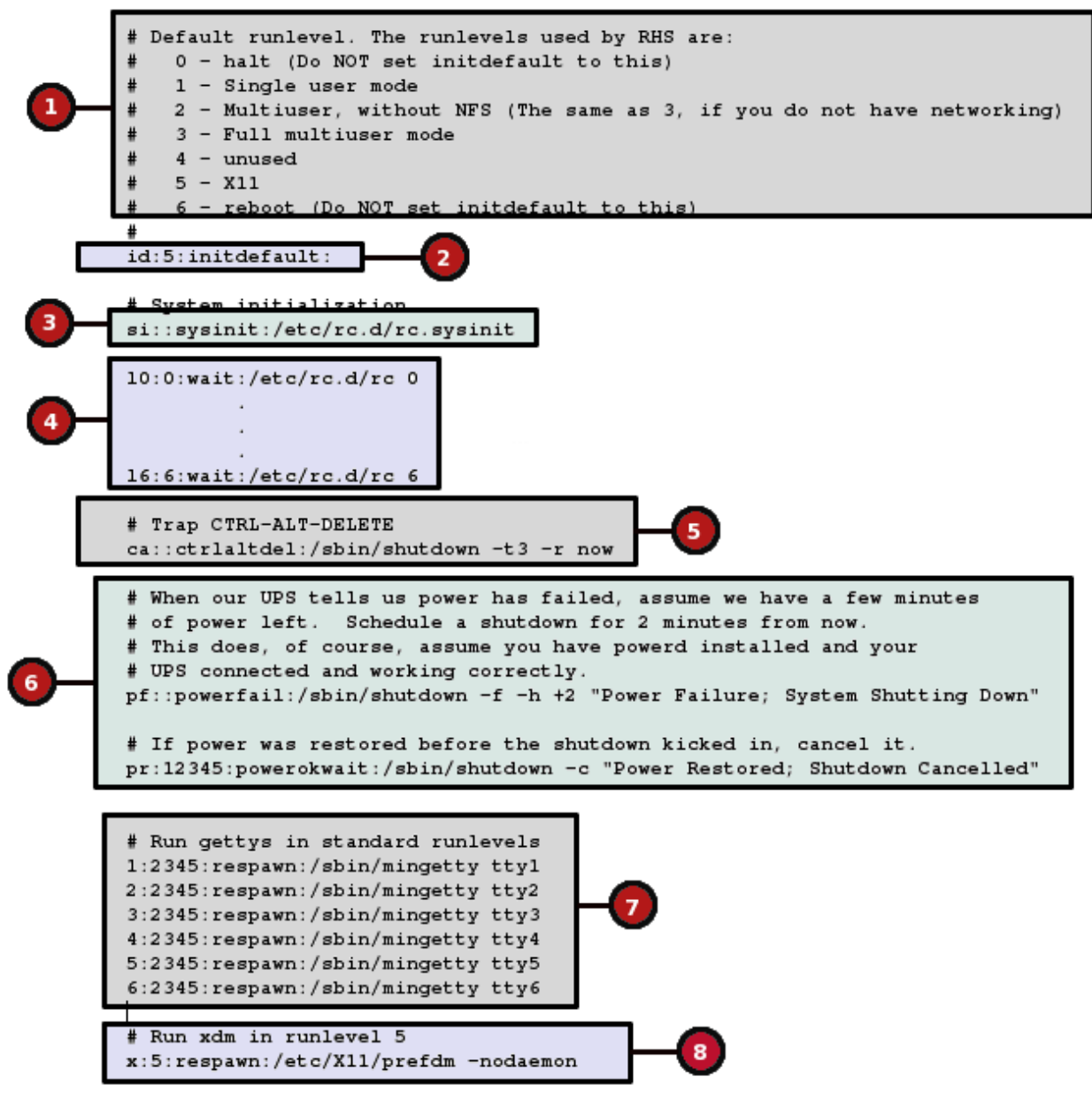
La ubicación de este fichero la podemos localizar en:



El fichero inittab describe que procesos se inician en la carga así como los scripts de inicialización del sistema, también distingue los múltiples niveles de ejecución bajo la cual trabaja el sistema operativo, recordemos que los niveles de ejecución válidos son 8, de los cuales tres son reservados y otro más es alias de algún nivel en particular.

De acuerdo a lo mostrado en la siguiente imagen describiremos la función que desempeña cada línea.

Para ello usamos como ejemplo el fichero inittab del sistema operativo Centos versión 5.2



1. Este recuadro nos indica los diferentes niveles de ejecución bajo los cuales trabaja CentOS 5.2
2. Este es el nivel de ejecución en el cual arranca por defecto el equipo
3. Aquí se especifica que script de configuración se debe cargar para el proceso de arranque del sistema
4. Aquí se especifican los scripts de arranque que el sistema utilizará para cada nivel de ejecución o también llamados runlevels
5. Esta sección controla el reseteo del sistema. Se puede comentar esta línea para que no pueda ser reseteado el sistema
6. Estas dos líneas están relacionadas con las acciones que deben seguirse en caso de una falla de voltaje y la segunda indica que debe hacerse cuando el voltaje ha sido restablecido.

7. Lineas encargadas de controlar los procesos getty
8. Esta linea arranca el entorno grafico del sistema, las cuales pueden ser GNOME o KDE

6.6 El fichero rc.sysinit

Este fichero esta localizado en la siguiente ruta

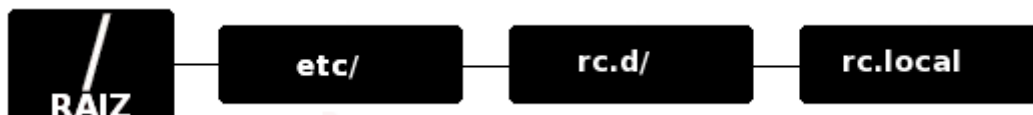


La funcion que desempeña este fichero es ejecutar una serie de scripts que inicializan tareas como:

- Configuracion de reloj del sistema
- Configuracion de los parametros del Kernel
- Levantamiento de dispositivos RAID y LVM
- Activacion y Actualizacion de cuotas en disco
- Activacion de la particion SWAP

6.7 El fichero rc.local

Este fichero esta localizado en la siguiente ruta



Este fichero es el ultimo en ser ejecutado por el proceso **init**.

La funcion que tiene este script es agregar comandos que nos haga facil de realizar tareas necesarias como arrancar servicios especiales o inicializar dispositivos sin tener que escribir scripts complejos de inicializacion en el directorio `/etc/rc.d/init.d` ni creando enlaces simbolicos.

6.8 Niveles de Ejecucion

Los niveles de ejecucion o tambien llamados runlevels hacen referencia a los sistemas operativos Linux que implementan el estilo de sistema de arranque de iniciacion tipo UNIX System V del cual ya hemos hablando extensamente.

A lo largo de este tema mencionamos 8 niveles de ejecucion bajo los cuales trabajan algunos sistemas operativos linux, en especial los de la Familia Red Hat. A continuacion mencionaremos las caracteristicas de cada uno de ellos.

6.8.1 Nivel 0 –Parada Del Sistema

El nivel 0 es usado para especificarle al sistema que debe apagarse, la forma en que este lo hace es a travez del comando halt.

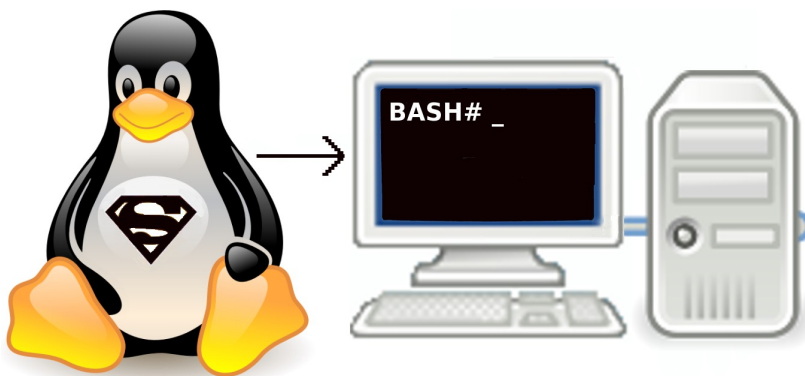
Al ejecutarse este comando se apagan todos los servicios que se encuentren activos



6.8.2 Nivel 1 o S -Monousuario o Single User

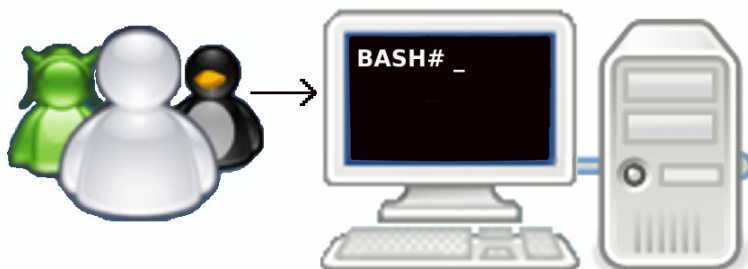
El nivel 1 o tambien llamado nivel Single (S) solo puede ser iniciado por el administrador del sistema (root), por lo que ningun usuario podra hacer eso de este nivel de ejecucion

En este nivel no se activan los servicios de Red, y tampoco se inician los procesos (daemons) de inicio por lo que permite reparar problemas o hacer pruebas al sistema.



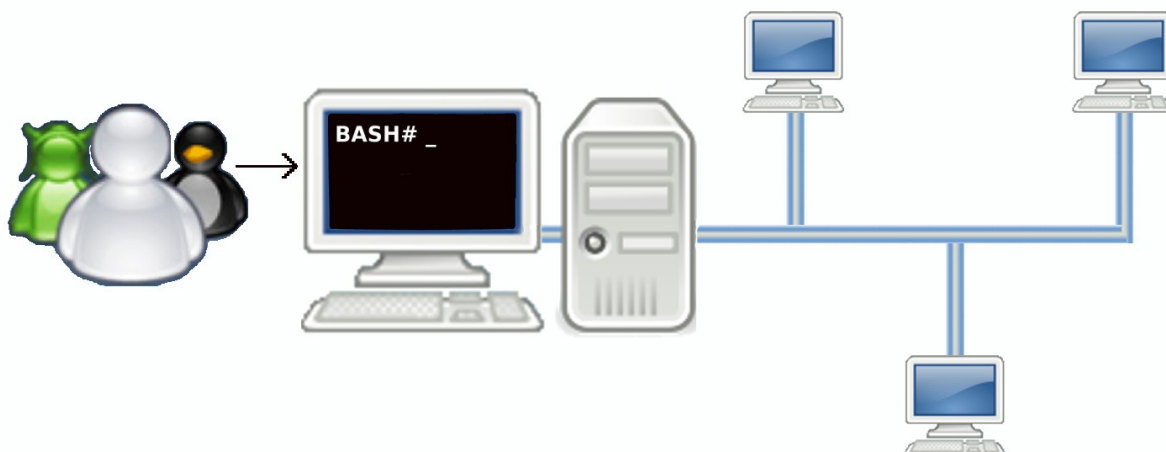
6.8.3 Nivel 2 -Multiusuario sin Red

Este nivel esta caracterizado por la capacidad de permitir que varios usuarios puedan entrar al sistema pero sin contar con soporte en red, esto quiere decir que no se puede contar con servidores como NFS o web.



6.8.4 Nivel 3 -Multiusuario con Red

Este sistema esta caracterizado por la capacidad de permitir a varios usuarios entrar al sistema, a diferencia del nivel de ejecucion 2, este si cuenta con soporte de red.

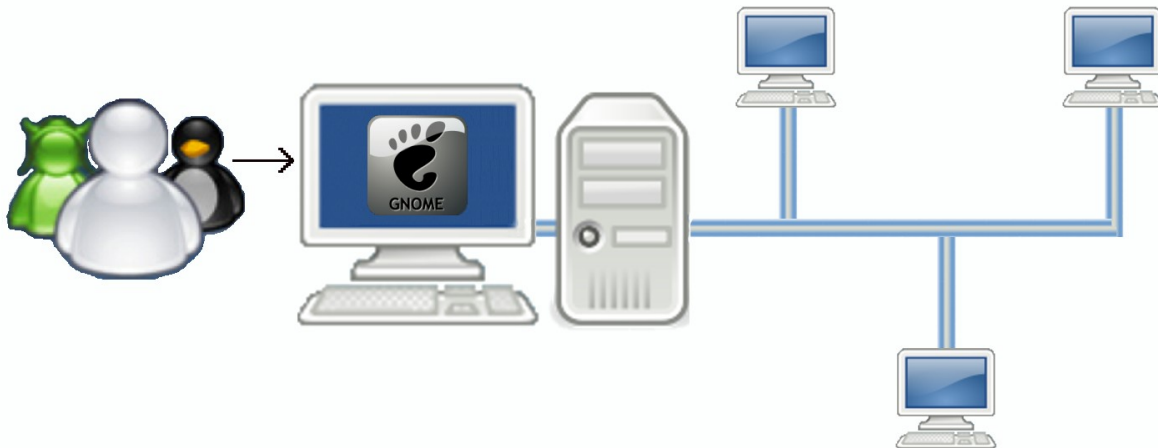


6.8.5 Nivel 4. -Sin Uso

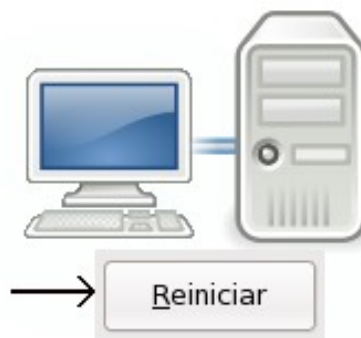
Para la mayoría de las distribuciones linux este nivel de ejecucion no tiene asignada ninguna funcion, pero puede ser personalizado por el administrador para que cumpla con alguna funcion en especial

6.8.6 Nivel 5. -Multiusuario Grafico

Este nivel de ejecucion es identico al nivel 3, la unica diferencia es el alta de entornos graficos como GNOME o KDE para la administracion del sistema



6.8.7 Nivel 6. -Reinicio del Sistema



6.9 Comando chkconfig

Este comando es una herramienta util para levantar o desactivar servicios los cuales son aplicados durante el arranque del equipo asi como tambien conocer el estado de los servicios que se estan ejecutando.

- Para conocer el estado de los procesos que estan corriendo en su sistema puede ejecutar el siguiente comando

```
[root@localhost ~]#chkconfig --list
```

- Para conocer el status de algun proceso en particular solo teclee esto

```
[root@localhost ~]#chkconfig --list httpd
httpd          0:desactivado    1:desactivado    2:desactivado
3:desactivado  4:desactivado    5:desactivado    6:desactivado
```

Lo cual nos mostrara bajo que niveles de ejecucion esta corriendo el proceso o servicio.

- Para levantar algun proceso o servicio durante el arranque del sistema solo teclee esto

```
[root@localhost ~]#chkconfig --level 35 httpd on
```

De esta manera estamos especificando el sistema que siempre que este inicie levante el servidor web apache en los niveles de ejecucion 3 y 5

- Para detener algun proceso o servicio durante el arranque del sistema solo teclee esto

```
[root@localhost ~]#chkconfig --level 35 httpd off
```

De esta manera estamos especificando el sistema que siempre que este inicie tenga detenido el servidor web apache en los niveles de ejecucion 3 y 5

6.10 Levantando, deteniendo y reiniciando servicios

Otra forma de levantar, detener o reiniciar servicios en caliente es mediante el uso del siguiente comando el cual hace uso del fichero init.d del cual hemos hablado anteriormente.

La estructura de la sintaxis para poder ocupar el comando es la siguiente:

```
[root@localhost ~]#/etc/init.d/nombreDelServicio {start|stop|status|
restart|reload}
```


A manera de ejemplificar el uso del anterior comando haremos lo siguiente.

Supongamos que tenemos ya instalado y configurado un servidor web apache y lo unico que falta es levantar el servicio, para ello solo bastara teclear lo siguiente:

```
[root@localhost ~]#/etc/init.d/httpd start
```

Para detener este mismo servicio solo debemos cambiar la palabra **start** por **stop**

```
[root@localhost ~]#/etc/init.d/httpd stop
```

De igual manera si se quiere reiniciar el servicio solo debemos cambiar la palabra **stop** por **restart**

```
[root@localhost ~]#/etc/init.d/httpd restart
```

Otra manera de arrancar, detener o reiniciar servicios sin necesidad de teclear toda la ruta anterior es mediante el uso de un alias el cual tiene la siguiente sintaxis

```
[root@localhost ~]# service httpd {start|stop|status|restart|reload}
```

De igual forma solo debemos teclear start, stop o restart según sea el caso

Ejemplo

```
[root@localhost ~]#service httpd start
```