

## **LPIC-2 / Examen 202 - Arranque del Sistema**

Este examen (en la nomenclatura oficial de LPI, algunos de estos objetivos están bajo el Examen 201 y otros bajo el 202) cubre el proceso de arranque, los gestores de arranque y la gestión de dispositivos con udev.

### **202.1 Personalizar el arranque del sistema con sysvinit**

#### **Teoría**

El proceso de arranque de un sistema Linux culmina cuando el kernel carga y ejecuta el primer proceso en el espacio de usuario. Este proceso, que siempre tiene el PID (Process ID) 1, es el sistema de inicio o "init system". Su responsabilidad es arrancar todos los demás servicios y procesos necesarios para que el sistema alcance un estado funcional.

#### **SysVinit (System V Init):**

SysVinit es un sistema de inicio tradicional, derivado del sistema de inicio de Unix System V. Fue el estándar en Linux durante muchos años antes de la llegada de sistemas más modernos como Upstart y systemd.

##### **1. El Proceso `init` (PID 1):**

- Es el padre de todos los demás procesos en el sistema.
- En SysVinit, este proceso es el ejecutable `/sbin/init`.

##### **2. Niveles de Ejecución (Runlevels):**

- SysVinit define diferentes "niveles de ejecución" (runlevels) que representan estados predefinidos del sistema.
- **Runlevels Estándar:**
  - 0: Halt (Apagar el sistema).
  - 1 (o S, s): Single-User Mode (Modo monousuario, para mantenimiento y reparación; solo se monta la raíz, no hay red ni la mayoría de los servicios).
  - 2: Multi-User Mode (Modo multiusuario sin servicios de red exportados - común en algunas distribuciones de escritorio antiguas).
  - 3: Multi-User Mode with Networking (Modo multiusuario completo con red - común en servidores).
  - 4: No utilizado por defecto, disponible para personalización.
  - 5: Multi-User Mode with Networking and Graphical Display Manager (Modo multiusuario completo con inicio de sesión gráfico - común en sistemas de escritorio).
  - 6: Reboot (Reiniciar el sistema).

##### **3. `/etc/inittab`:**

- El archivo de configuración principal de SysVinit. Define el comportamiento de `init`.
- **Directivas Clave (aunque a menudo ignoradas por systemd):**

- `id:runlevel:initdefault::` Define el nivel de ejecución por defecto al arrancar.
- `lX:runlevels:action:process:` Define procesos que se ejecutan en terminales (TTYs) para iniciar sesión (ej: `l1:2345:respawn:/sbin/agetty tty1 ...`).
- `rX:runlevels:action:process:` Define los scripts (jobs) que se ejecutan al entrar o salir de un runlevel.
- **Nota:** En sistemas con `systemd`, el archivo `/etc/inittab` es a menudo ignorado por completo, o solo se usa la línea `initdefault` por compatibilidad. La configuración del runlevel por defecto y los servicios se maneja de otra manera en `systemd`.

#### 4. Scripts de Inicio (`/etc/init.d/`):

- Directorio que contiene los scripts de shell tradicionales para controlar los servicios del sistema.
- Cada script (`apache2`, `ssh`, `networking`, etc.) debe aceptar argumentos estándar: `start`, `stop`, `restart`, `status`, `reload`.
- `sudo /etc/init.d/<nombre_servicio> start|stop|restart|status`: Inicia, detiene, reinicia o verifica el estado de un servicio manualmente.

#### 5. Directorios de Runlevels (`/etc/rc[0-6].d/`):

- SysVinit utiliza estos directorios (uno por runlevel) para determinar qué scripts de `/etc/init.d/` deben ejecutarse al entrar o salir de un runlevel.
- Estos directorios contienen **enlaces simbólicos** que apuntan a los scripts originales en `/etc/init.d/`.
- Los nombres de los enlaces simbólicos determinan si el script debe iniciarse o detenerse y el orden:
  - Nombres que empiezan con S (Start): El script se inicia al entrar en este runlevel.
  - Nombres que empiezan con K (Kill): El script se detiene al entrar en este runlevel.
  - Los dos dígitos después de S o K (##): Definen el orden de ejecución (números más bajos se ejecutan antes).
  - Ejemplo: `/etc/rc3.d/S20ssh` es un enlace a `/etc/init.d/ssh`. Significa que al entrar en el runlevel 3, se ejecutará `/etc/init.d/ssh start` con un orden de ejecución 20. `/etc/rc0.d/K01ssh` significa que al entrar en el runlevel 0 (halt), se ejecutará `/etc/init.d/ssh stop` con un orden 01.
- Herramientas como `update-rc.d` (Debian) o `chkconfig` (Red Hat) gestionan la creación/eliminación de estos enlaces simbólicos.

#### 6. Cambio de Runlevel:

- `runlevel`: Muestra el runlevel anterior y el runlevel actual. La primera letra puede ser N si no ha habido cambio desde el arranque.
- `sudo init <runlevel>`: Cambia el sistema al runlevel especificado. **¡Usar con extrema precaución!** `init 0` apaga el sistema, `init 6` reinicia, `init 1` va a modo monousuario (lo que puede cerrar tu sesión si no estás en la consola local).

#### Diferencias Debian vs. Red Hat (SysVinit):

- La estructura de directorios (`/etc/init.d/`, `/etc/rcX.d/`, `/etc/inittab`) es la misma, ya que son convenciones de SysVinit.
- Las herramientas para gestionar los enlaces simbólicos varían: `update-rc.d` en Debian/Ubuntu, `chkconfig` en Red Hat/CentOS/Fedora (aunque `chkconfig` también gestionaba SysVinit scripts antes de `systemd`).
- El contenido por defecto de los scripts individuales en `/etc/init.d/` puede variar.
- Lo más importante: En las distribuciones modernas, el sistema **no** usa SysVinit por defecto, sino `systemd`. El proceso `init` (PID 1) es `/sbin/init` que es un enlace simbólico a `/lib/systemd/systemd`. Aunque los scripts en `/etc/init.d/` pueden seguir existiendo (por compatibilidad) y `systemd` puede ejecutarlos, la lógica principal del arranque y los "targets" de `systemd` (que reemplazan a los runlevels) son diferentes. LPIC-2 requiere entender SysVinit, pero es probable que tu VM esté usando `systemd`.