

## 104.7 Encontrar archivos del sistema y ubicarlos en el lugar correcto - Ejercicios

### Ejercicio 4.7.1: Usando **which** y **whereis** para Encontrar Comandos

- **Objetivo:** Localizar ejecutables, fuentes y páginas de manual de comandos.
- **Desarrollo Paso a Paso:**
  1. Abre una terminal.
  2. **Encuentra la ubicación del ejecutable `cat`:** Ejecuta `which cat`. Te mostrará la ruta (ej: `/usr/bin/cat`).
  3. **Encuentra el ejecutable y la página de manual de `ls`:** Ejecuta `whereis ls`. Te mostrará la ruta del binario y de la página de manual comprimida.
  4. **Encuentra un comando que pueda estar en `/sbin` (requiere `sudo` para algunos **which**):** Ejecuta `which fdisk`. Puede que no lo encuentre porque `/sbin` no siempre está en el PATH de un usuario normal. Ejecuta `sudo which fdisk`. Ahora debería mostrarte la ruta (ej: `/sbin/fdisk`).
  5. **Encuentra binario, fuente y man de `find`:** Ejecuta `whereis find`.

### Ejercicio 4.7.2: Usando **locate** para Búsqueda Rápida (Basada en DB)

- **Objetivo:** Buscar archivos utilizando la base de datos de `locate` y actualizarla.
- **Requisitos:** El paquete `mlocate` o `findutils` (que contiene `locate`) debe estar instalado.
- **Desarrollo Paso a Paso:**
  1. Abre una terminal.
  2. **Busca un archivo conocido:** Ejecuta `locate resolv.conf`. Debería mostrar la ruta completa a `/etc/resolv.conf`.
  3. **Busca todos los archivos `.log` en el sistema:** Ejecuta `locate "*.log"`. La salida puede ser muy larga. Presiona `Ctrl+C` para detenerla.
  4. **Crea un archivo nuevo:** Ejecuta `touch nuevo_archivo_locate.txt`.
  5. **Intenta buscar el nuevo archivo con `locate`:** Ejecuta `locate nuevo_archivo_locate.txt`. Probablemente no lo encontrará porque la base de datos no está actualizada.
  6. **Actualiza la base de datos (requiere `sudo`):** Ejecuta `sudo updatedb`. Esto puede tardar unos segundos o minutos dependiendo del tamaño de tu sistema de archivos.
  7. **Busca el nuevo archivo de nuevo:** Ejecuta `locate nuevo_archivo_locate.txt`. Ahora sí debería encontrarlo.
  8. **Limpia:** Ejecuta `rm nuevo_archivo_locate.txt`. Después de la próxima ejecución de `updatedb`, el archivo desaparecerá de la base de datos.

### Ejercicio 4.7.3: Usando **find** para Búsqueda Detallada (En Tiempo Real)

- **Objetivo:** Buscar archivos por varios criterios utilizando `find`.

- **Desarrollo Paso a Paso:**

1. Abre una terminal.
2. **Crea algunos archivos de prueba con diferentes nombres, tipos y permisos:**
  - `mkdir find_test_dir`
  - `touch find_test_dir/file1.txt`  
`find_test_dir/file2.log find_test_dir/script.sh`
  - `mkdir find_test_dir/sub_dir`
  - `chmod 755 find_test_dir/script.sh`
  - `chmod 600 find_test_dir/file2.log`
3. **Busca archivos por nombre en el directorio actual y subdirectorios:** Ejecuta `find find_test_dir -name "file1.txt"`.
4. **Busca archivos por patrón de nombre (terminados en .log):** Ejecuta `find find_test_dir -name "*.log"`.
5. **Busca directorios por nombre:** Ejecuta `find find_test_dir -type d -name "sub_dir"`.
6. **Busca archivos con permisos específicos (ej: 755):** Ejecuta `find find_test_dir -perm 755`. Debería encontrar `find_test_dir/script.sh` y `find_test_dir/sub_dir`. (Nota: `find` interpreta permisos de formas complejas; `-perm mode` busca archivos con permisos *exactamente* mode; `-perm /mode` busca archivos donde *cualquiera* de los bits del mode esté activado; `-perm -mode` busca archivos donde *todos* los bits del mode estén activados). Una búsqueda más común es `find . -type f -perm 644` para archivos regulares con permisos 644.
7. **Busca archivos por usuario propietario:** Ejecuta `find find_test_dir -user tu_usuario`. Encontrará todos los archivos que creaste.
8. **Busca archivos modificados recientemente (ej: en los últimos 60 minutos):** Ejecuta `find find_test_dir -mmin -60`.
9. **Busca archivos mayores que un tamaño específico (ej: 1KB):** Crea un archivo grande: `dd if=/dev/zero of=find_test_dir/large_file bs=1k count=2`. Ejecuta `find find_test_dir -size +1k`. Busca archivos con tamaño exactamente 2k: `find find_test_dir -size 2k`. Busca archivos con tamaño mayor o igual a 2MB: `find find_test_dir -size +2M`.
10. **Encuentra archivos y ejecuta una acción sobre ellos (ej: listar):** Ejecuta `find find_test_dir -name "*.txt" -ls`.
11. **Limpia:** Ejecuta `rm -r find_test_dir`.

#### Ejercicio 4.7.4: Explorando la Jerarquía del Sistema de Archivos (FHS)

- **Objetivo:** Familiarizarse con el contenido típico de los directorios clave del FHS.
- **Desarrollo Paso a Paso:**
  1. Abre una terminal.

2. **Explora el directorio raíz:** Ejecuta `ls -l /`. Identifica los subdirectorios clave (`bin`, `etc`, `home`, `usr`, `var`, etc.).
3. **Explora los binarios esenciales:** Ejecuta `ls /bin`. Verás comandos como `cat`, `ls`, `mv`, `rm`, `date`.
4. **Explora los binarios del sistema esenciales:** Ejecuta `ls /sbin`. Verás comandos como `fdisk`, `fsck`, `mkfs`, `mount`. (Nota: En muchas distribuciones modernas, `/bin` y `/sbin` son enlaces simbólicos a `/usr/bin` y `/usr/sbin` respectivamente, como parte de una unificación. Puedes verificarlo con `ls -l /`).
5. **Explora archivos de configuración:** Ejecuta `ls /etc`. Este es un directorio muy grande. Busca archivos conocidos como `passwd`, `group`, `fstab`, o directorios de configuración como `ssh`, `apt`.
6. **Explora los binarios no esenciales:** Ejecuta `ls /usr/bin`. Este es aún más grande que `/bin`. Contiene la mayoría de las aplicaciones de usuario.
7. **Explora el directorio de logs:** Ejecuta `ls /var/log`. Verás archivos de log como `syslog`, `auth.log`, `kern.log`, `dmesg`.
8. **Explora un directorio temporal:** Ejecuta `ls /tmp`.