

2.4 Crear, Mover y Eliminar Archivos y Directorios

Introducción: Una vez que puedes navegar por el sistema de archivos, el siguiente paso es aprender a manipular su contenido: crear nuevos archivos y directorios, copiarlos, moverlos, renombrarlos y eliminarlos.

Comandos Esenciales:

1. **touch:**

- **Uso:** Principalmente, actualiza la fecha de último acceso y modificación de un archivo existente a la hora actual. Si el archivo no existe, lo crea vacío.
- **Sintaxis:** `touch [opciones] nombre_archivo...`
- **Ejemplos:**
 - `touch mi_archivo.txt`: Crea `mi_archivo.txt` si no existe, o actualiza su fecha si ya existe.
 - `touch archivo1.txt archivo2.log`: Puede crear/actualizar varios archivos a la vez.

2. **mkdir (Make Directory):**

- **Uso:** Crea uno o más directorios nuevos.
- **Sintaxis:** `mkdir [opciones] nombre_directorio...`
- **Ejemplos:**
 - `mkdir mi_directorio`: Crea un directorio llamado `mi_directorio` dentro del directorio actual.
 - `mkdir dir1 dir2 dir3`: Crea tres directorios.
 - `mkdir -p ruta/larga/de/directorios`: La opción `-p` (parents) es muy útil, ya que crea los directorios padres intermedios si no existen. Sin `-p`, si `ruta` o `larga` no existen, el comando fallaría.

3. **cp (Copy):**

- **Uso:** Copia archivos o directorios.
- **Sintaxis:**
 - `cp [opciones] origen destino` (copiar un archivo a otro archivo o a un directorio)
 - `cp [opciones] origen... directorio_destino` (copiar múltiples archivos a un directorio)
- **Ejemplos:**
 - `cp archivo1.txt archivo1.bak`: Copia `archivo1.txt` a `archivo1.bak` en el mismo directorio.
 - `cp main.c ../backup/`: Copia `main.c` al directorio `backup` que está un nivel arriba.
 - `cp /etc/hosts .`: Copia el archivo `/etc/hosts` al directorio actual (`.`).

- `cp *.txt documentos/`: Copia todos los archivos `.txt` al subdirectorio `documentos`.
- `cp -r directorio1 directorio_destino`: Para copiar un directorio y todo su contenido, es **obligatorio** usar la opción `-r` (recursive).
- **Opciones Comunes:**
 - `-r` o `-R` (recursive): Copia directorios y su contenido recursivamente.
 - `-i` (interactive): Pregunta antes de sobrescribir un archivo existente en el destino.
 - `-v` (verbose): Muestra qué archivos se están copiando.
 - `-p` (preserve): Conserva atributos como permisos, propietario y fechas de modificación.

4. **mv (Move):**

- **Uso:** Mueve archivos o directorios de un lugar a otro. También se usa para **renombrar** archivos o directorios si el origen y el destino están en el mismo directorio.
- **Sintaxis:**
 - `mv [opciones] origen destino` (mover/renombrar un archivo/directorio)
 - `mv [opciones] origen... directorio_destino` (mover múltiples archivos/directorios a un directorio)
- **Ejemplos:**
 - `mv archivo_viejo.txt archivo_nuevo.txt`: Renombra `archivo_viejo.txt` a `archivo_nuevo.txt`.
 - `mv reporte.pdf ../informes/`: Mueve `reporte.pdf` al directorio `informes` que está un nivel arriba.
 - `mv dir1 dir2 /tmp/`: Mueve los directorios `dir1` y `dir2` (y su contenido) al directorio `/tmp`.
- **Opciones Comunes:**
 - `-i` (interactive): Pregunta antes de sobrescribir.
 - `-v` (verbose): Muestra qué se está moviendo/renombrando.

5. **rm (Remove):**

- **Uso:** Elimina archivos. **¡Este comando es peligroso!** Los archivos eliminados con `rm` generalmente no se pueden recuperar fácilmente (no van a una "papelera"). Úsalo con precaución.
- **Sintaxis:** `rm [opciones] nombre_archivo...`
- **Ejemplos:**
 - `rm archivo_temporal.tmp`: Elimina el archivo `archivo_temporal.tmp`.
 - `rm *.log`: Elimina todos los archivos que terminan en `.log`.

- **Opciones Comunes:**

- `-i` (interactive): Pregunta antes de eliminar cada archivo. **Recomendado para principiantes.**
- `-f` (force): Fuerza la eliminación sin preguntar, incluso si los archivos son de solo lectura. **Muy peligroso, especialmente combinado con `-r`.**
- `-r` o `-R` (recursive): Elimina directorios y todo su contenido recursivamente. **Extremadamente peligroso si se usa incorrectamente (e.g., `rm -rf /`).**
- `-v` (verbose): Muestra qué archivos se están eliminando.

6. `rmdir` (Remove Directory):

- **Uso:** Elimina directorios, **pero solo si están vacíos.**
- **Sintaxis:** `rmdir [opciones] nombre_directorio...`
- **Ejemplo:** `rmdir directorio_vacio`
- **Nota:** Para eliminar directorios con contenido, debes usar `rm -r`.

Precauciones:

- Siempre verifica en qué directorio estás (`pwd`) antes de usar `rm`, especialmente con comodines (`*`) u opciones recursivas (`-r`).
- Considera usar la opción `-i` con `cp`, `mv` y `rm` hasta que te sientas cómodo.
- Evita usar `rm -rf` a menos que estés absolutamente seguro de lo que estás haciendo. Un error tipográfico puede ser catastrófico.

Fuentes y Más Información:

- [Top 100 Comandos Linux \(DreamHost\)](#) (Cubre `mkdir`, `rmdir`, `cp`, `mv`, `rm`)
- [Los 40 comandos de Linux más utilizados que debe conocer \(Kinsta\)](#) (Explica `cp`, `rm`, `mv`)
- [41 comandos básicos para el terminal de Linux... \(Xataka\)](#) (Incluye `mkdir`, `rmdir`, `cp`, `mv`, `rm`, `touch`)
- [Comandos básicos para usar en el terminal \(SW Hosting\)](#) (Menciona `mkdir`, `rmdir`, `mv`, `rm`)
- [25 comandos de Linux básicos \[Para principiantes\] \(Axarnet\)](#) (Advierte sobre los peligros de `rm -rf`)