



Examen 103 - Comandos GNU y Unix

103.8 Realizar edición básica de archivos usando vi

Teoría

vi (Visual Editor) es un editor de texto modal creado a finales de los 70. Aunque su interfaz puede parecer arcaica comparada con editores modernos, es extremadamente potente, rápido y, lo más importante para LPIC, está disponible en prácticamente cualquier sistema Unix o Linux. **vim** (Vi IMproved) es una versión mejorada que mantiene compatibilidad con **vi** pero añade muchas características útiles (resaltado de sintaxis, deshacer multinivel, etc.). En la práctica, cuando ejecutas **vi**, a menudo se está ejecutando **vim** en modo compatibilidad.

Concepto Clave: Modos de Vi

La característica más importante y a menudo confusa de **vi** es su naturaleza *modal*. No siempre estás escribiendo texto directamente. **vi** tiene varios modos principales:

1. **Modo Normal (o Comando):** Es el modo por defecto al abrir **vi**. En este modo, las pulsaciones de teclas se interpretan como *comandos* para mover el cursor, borrar texto, copiar, pegar, cambiar modos, etc., *no* como caracteres a insertar en el texto.
 - **Para entrar:** Presiona la tecla **ESC**. Siempre que no estés seguro en qué modo te encuentras, presiona **ESC** un par de veces para volver al modo Normal.
2. **Modo Insertar:** En este modo, las pulsaciones de teclas se interpretan como caracteres que se insertan en el texto. Es el modo donde realmente escribes.
 - **Para entrar (desde Modo Normal):**
 - **i:** Insertar texto *antes* del cursor.
 - **a:** Insertar texto *después* del cursor (append).
 - **O:** Insertar una nueva línea *debajo* de la actual y entrar en modo Insertar.
 - **O:** Insertar una nueva línea *encima* de la actual y entrar en modo Insertar.
 - **I:** Insertar texto al *principio* de la línea actual.
 - **A:** Insertar texto al *final* de la línea actual.
 - **Para salir:** Presiona la tecla **ESC**.
3. **Modo Última Línea (o Ex Mode):** Se utiliza para ejecutar comandos más complejos, como guardar el archivo, salir, buscar y reemplazar (a menudo usando regex), configurar opciones, etc. Los comandos en este modo se introducen en la última línea de la pantalla.
 - **Para entrar (desde Modo Normal):** Presiona dos puntos (:). La barra / también te lleva a la última línea para comandos de búsqueda.
 - **Para salir:** Después de ejecutar el comando en la última línea (presionando Enter), vuelves automáticamente al Modo Normal.

Comandos Básicos (Desde Modo Normal):

- **Movimiento del Cursor:**
 - h: Mover cursor a la izquierda.
 - j: Mover cursor hacia abajo.
 - k: Mover cursor hacia arriba.
 - l: Mover cursor a la derecha.
 - w: Mover cursor al principio de la siguiente palabra.
 - b: Mover cursor al principio de la palabra anterior.
 - 0 (cero): Mover cursor al principio de la línea.
 - ^: Mover cursor al primer carácter no blanco de la línea.
 - \$: Mover cursor al final de la línea.
 - G: Mover cursor a la última línea del archivo.
 - gg: Mover cursor a la primera línea del archivo.
 - nM (donde n es un número): Ir a la línea n. Ej: 5G va a la línea 5.
- **Edición (desde Modo Normal):**
 - x: Borrar el carácter bajo el cursor.
 - dw: Borrar la palabra desde el cursor hasta el final de la palabra.
 - dd: Borrar la línea actual. Puedes prefijar con un número para borrar múltiples líneas (ej: 3dd borra la línea actual y las 2 siguientes).
 - yy: Copiar (yank) la línea actual. Puedes prefijar con un número (ej: 2yy copia 2 líneas).
 - p: Pegar (put) el contenido copiado o cortado *después* del cursor o línea actual.
 - P: Pegar *antes*.
 - u: Deshacer la última acción.
 - Ctrl+r: Rehacer la última acción deshecha (en vim).
- **Modo Última Línea (desde Modo Normal, presiona :):**
 - :w: Guardar el archivo.
 - :q: Salir del editor.
 - :wq o :x: Guardar y salir.
 - :q!: Salir sin guardar los cambios (descartar cambios).
 - :w <nombre_archivo>: Guardar el archivo con un nuevo nombre.
 - :! <comando_shell>: Ejecutar un comando de shell temporalmente. (ej: :! ls)
- **Búsqueda (desde Modo Normal):**
 - /patron: Buscar patron hacia adelante. Presiona n para la siguiente coincidencia, N para la anterior.
 - ?patron: Buscar patron hacia atrás.
 - *: Buscar la palabra bajo el cursor hacia adelante.
 - #: Buscar la palabra bajo el cursor hacia atrás.

Vim: Mejoras Comunes

- Resaltado de sintaxis.

- Deshacer/Rehacer persistente y multinivel.
- Autocompletado.
- Divisiones de ventana.
- Configuración extensa a través de `~/ .vimrc`.
- Soporte para plugins.

Aunque LPIC-1 se centra en `vi` básico, entender que `vim` es el que usarás en la práctica es importante. Los comandos básicos son los mismos.