

Supervisión de Procesos y Optimización del Rendimiento en Linux

Fase 1: Análisis en tiempo real

- Comandos utilizados:
 - `top` → Procesos activos, uso CPU/RAM.
 - `htop` → Visualización mejorada (instalado con `apt install htop`).
 - `uptime` → Tiempo encendido + carga media.
 - `free -m` → Uso de memoria RAM.
- Identificación:
 - Proceso con mayor consumo de CPU: obtenido desde `top` o `htop`.
 - Proceso con mayor uso de RAM: visto desde `htop`.
 - Tiempo de encendido y carga del sistema: consultado con `uptime`.

Fase 2: Gestión activa de procesos

- Eliminación de proceso innecesario:
 - `kill <PID>`
- Cambiar prioridad:
 - `renice +10 -p <PID>` → Bajar prioridad.
- Ejecución en segundo y primer plano:
 - `sleep 100 &` → Segundo plano.
 - `fg` → Pasar a primer plano.
- Ejecución con baja prioridad:
 - `nice -n 10 cp ~/Escritorio/prueba ~/Escritorio/1/`

Fase 3: Monitorización y registro

- `vmstat > /srv/logs/vmstat.log` → Registro de estado del sistema.
- Crontab configurado con:
 - `* /5 * * * * top -b -n 1 >> /srv/logs/top.log`
- Monitorización de disco:
 - `iotop` (instalado con `apt install iotop`).

Fase 4: Simulación de sobrecarga

- Herramienta instalada: stress-ng.
- Pruebas realizadas:
 - CPU: stress-ng --cpu 2 --timeout 60s
- Observación con htop: se comprobó el impacto de cada carga en tiempo real.

Resultado final

- Se ejecutaron correctamente todos los comandos.
- Se analizaron procesos y recursos.
- Se optimizó el sistema con control de prioridades y eliminación de procesos inactivos.
- Se dejó configurada una monitorización automática mediante crontab.