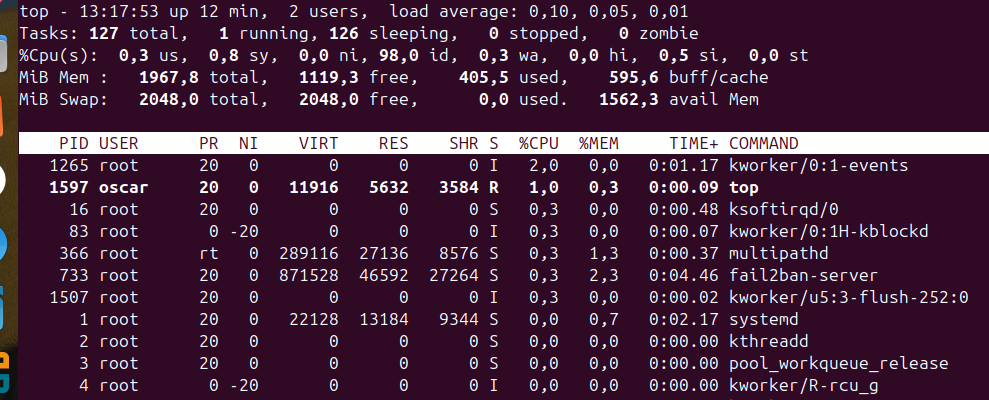
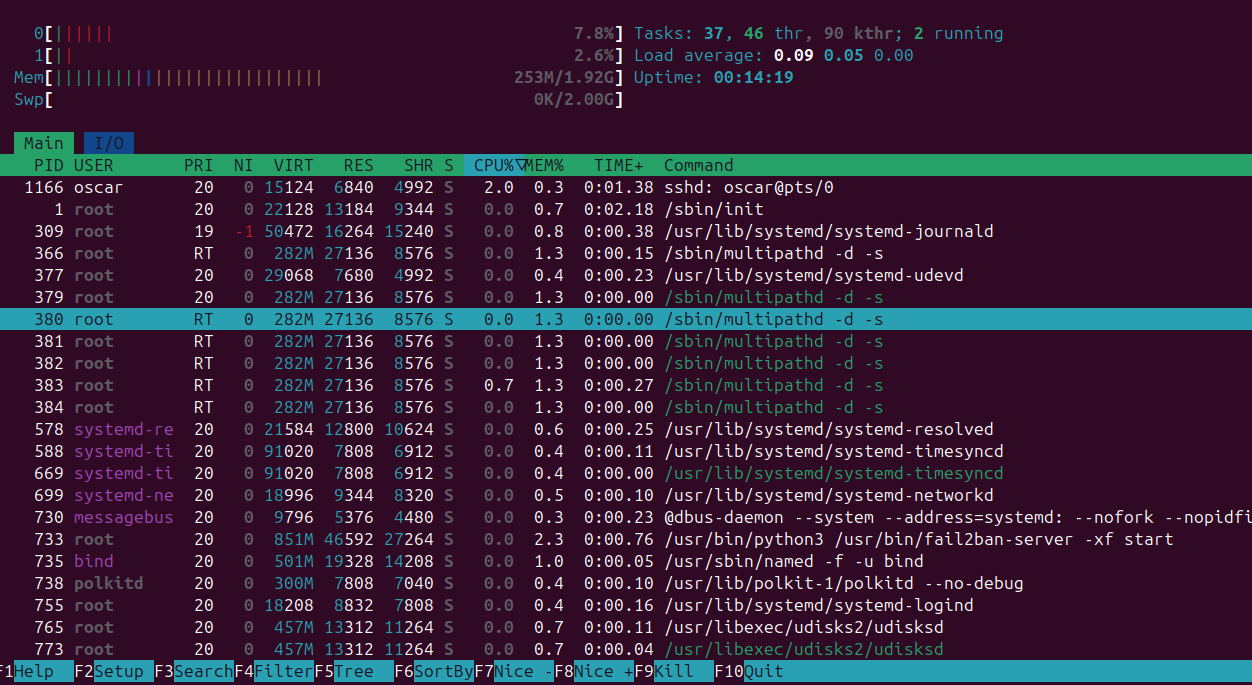
**fase 1, analisis en tiempo real del sistema**

ejecutar y analizar las herramientas, top y ver procesos activos en carga y uso de cpu, htop y uptime y free -m

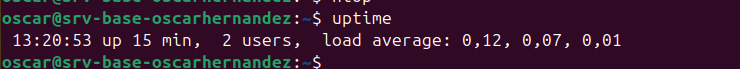
ejecutar top



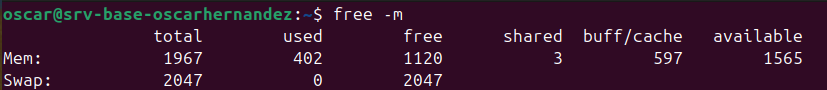
ejecutar htop



utime



free -m



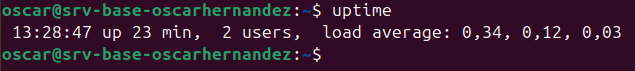
identificar el proceso con mayor consumo de CPU haciendo uso de top y p para ordenarlos



identificar el proceso con mayor uso de memoria con el comando top y pulsando m



identificar tipo que lleva el sistema encendido y carga promedio



lleva encendido 23 minutos hay dos usuario, que son el root y ssh porque estoy conectado con la reminal de ubuntu y la carga del sistem en el ultimo minuto es de 0.34, en los ultimos 5 minutos es de 0.12 y en los ultimos 15 minutos es de 0.03

**fase 2, gestion activa de procesos y prioridades**

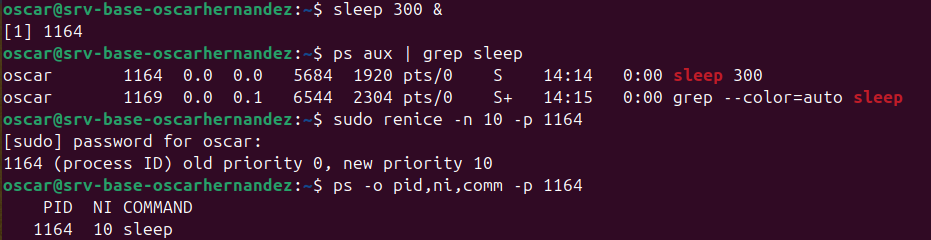
hacemos la prueba abriendo un terminal que no afecte a nada



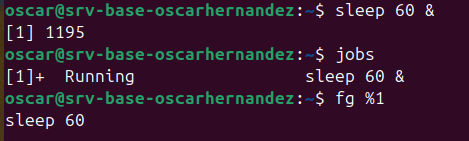
y terminamos el proceso con pkil



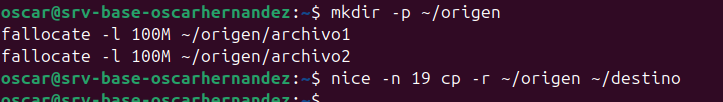
Cambiar la prioridad de un proceso en ejecución con renice, abrimos un proceso nuevo para que no afecte a al sistema con sleep 300 &, miramos el pid con ps aux | grep sleep y cogemos el pid que es 1164, y cambiamos la prioridad del proceso con sudo renice -n 10 -p 1164 por ultimo miramos si se ha cambiado la prioridad con ps -o pid,ni,comm -p 1164



lanzar un proceso en segundo plano y enviarlo a primer plano con fg



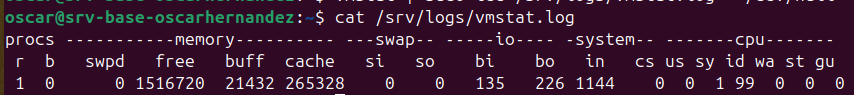
Usar nice para iniciar un proceso con prioridad baja (por ejemplo, una copia pesada con cp). creamos dos archivos y usamos nice para hacer un proceso prioridad baja de copia de esos dos archivos



**fase 3, monitorización y registro de uso de recursos**

Usar el comando vmstat y guardar su salida en un archivo /srv/logs/vmstat.log.

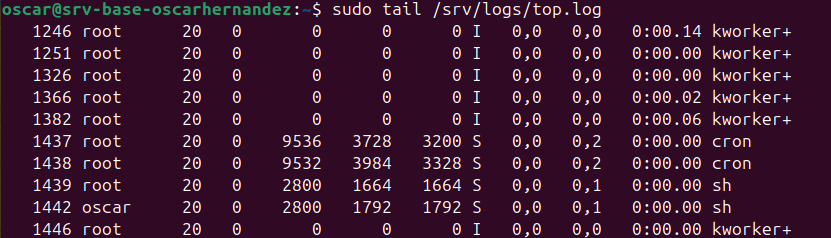
primero creamos el archivo y comprobamos que funciona



en el archo de configuración añadimos



por ultimo comprobar si funcionna correctamente

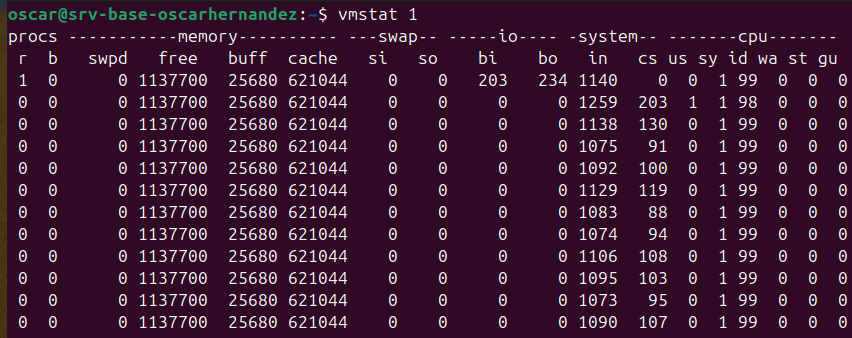


**fase 4, simulación de sobrecarga simulado**

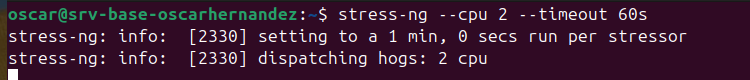
Instalar el paquete stress o stress-ng.



Ejecutar una prueba con carga simulada de CPU, primero miramos el rendimiento de cpu



luego ejecutamos la prueba durante 60 segundos



y esperamos los resultados

