

## Данные о начальной конфигурации операционной системы в аспекте управления памятью

- 1) Общий объем оперативной памяти = 3998740 kB
- 2) Объем раздела подкачки = 3997692 kB
- 3) Размер страницы виртуальной памяти = 4096 B
- 4) Объем свободной физической памяти в ненагруженной системе = 1943552 kB
- 5) Объем свободного пространства в разделе подкачки в ненагруженной системе = 3997692 kB

### Эксперимент №1

*Первый этап*

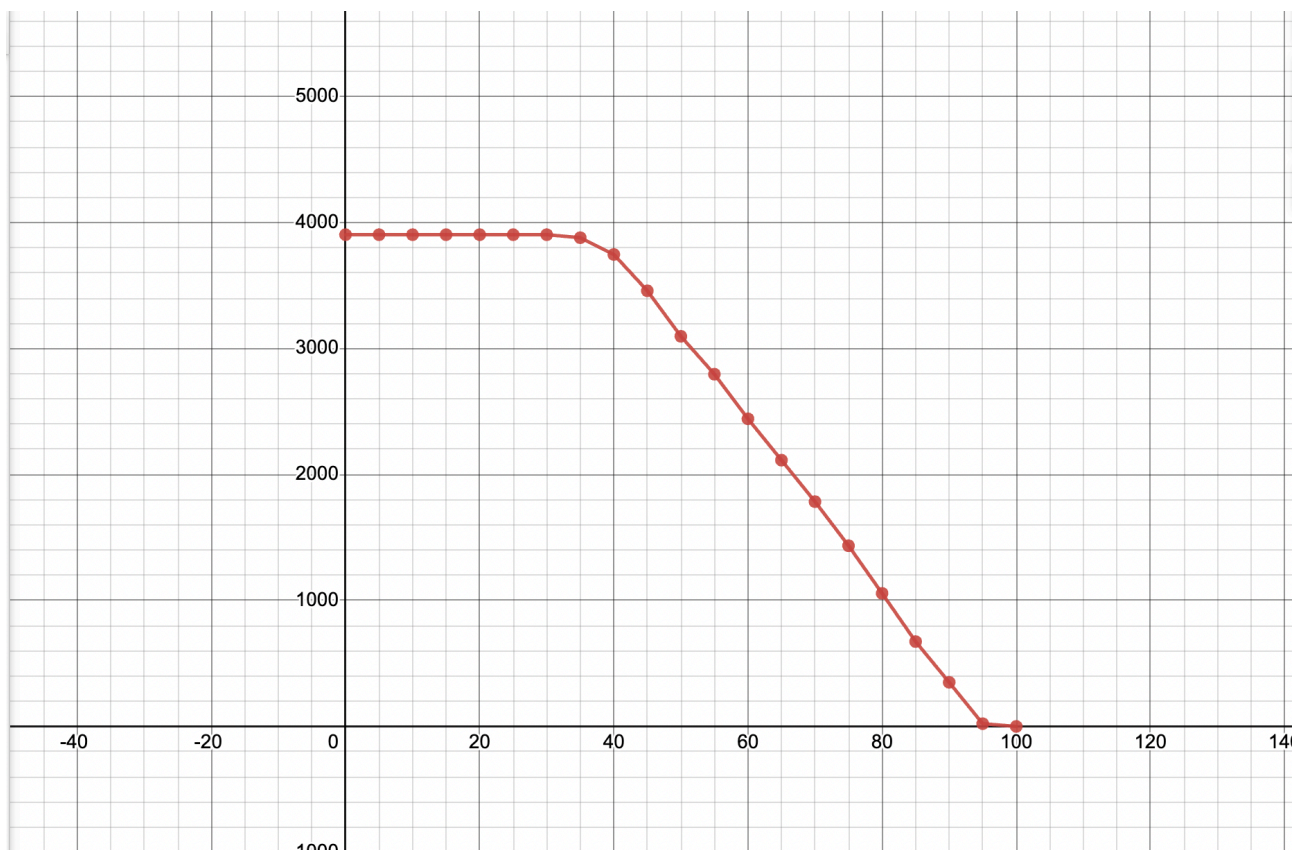
### Последние записи журнала

[ 164.651558] oom-kill:constraint=CONSTRAINT\_NONE,nodemask=(null),cpuset=user.slice,mems\_allowed=0,global\_oom,task\_memcg=/user.slice/user-1000.slice/user@1000.service/app.slice/app-org.gnome.Terminal.slice/vte-spawn-feaf12de-299e-4ea2-a984-e276c4317f83.scope,task=mem.sh,pid=2383,uid=1000

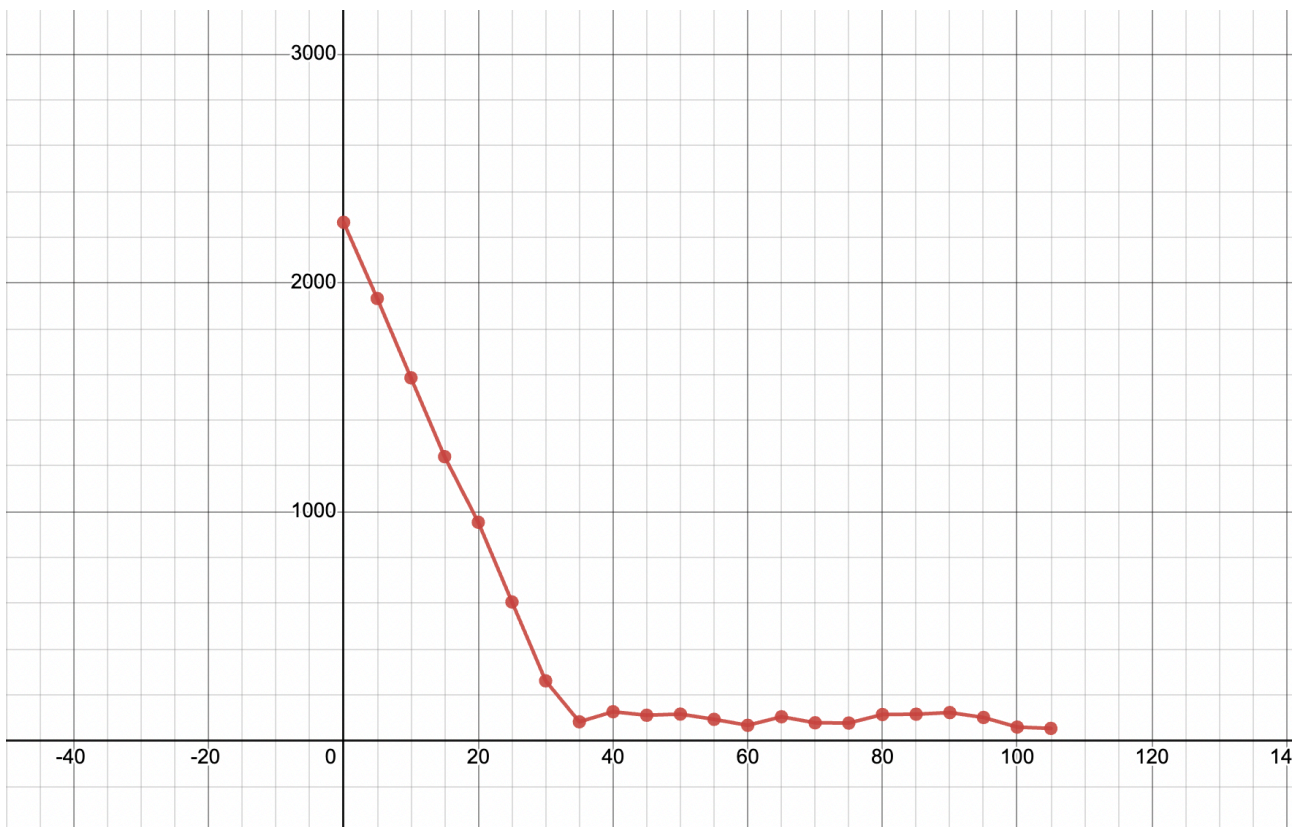
[ 164.651565] Out of memory: Killed process 2383 (mem.sh) total-vm:6953948kB, anon-rss:3604224kB, file-rss:1536kB, shmem-rss:0kB, UID:1000 pgtables:13644kB oom\_score\_adj:200

**Значение в последней строке файла report.log = 88000000**

**График зависимости swar (подкачки) от времени**



**График зависимости mem (свободной памяти) от времени**



**Изменения в верхних пяти процессах лежат в top.log. Лидирующую позицию занимает mem.sh**

## Вывод

По графикам можно понять, когда оперативная память оказывается недостаточной для активных процессов, операционная система использует раздел подкачки, чтобы временно освободить физическую память.

### *Второй этап*

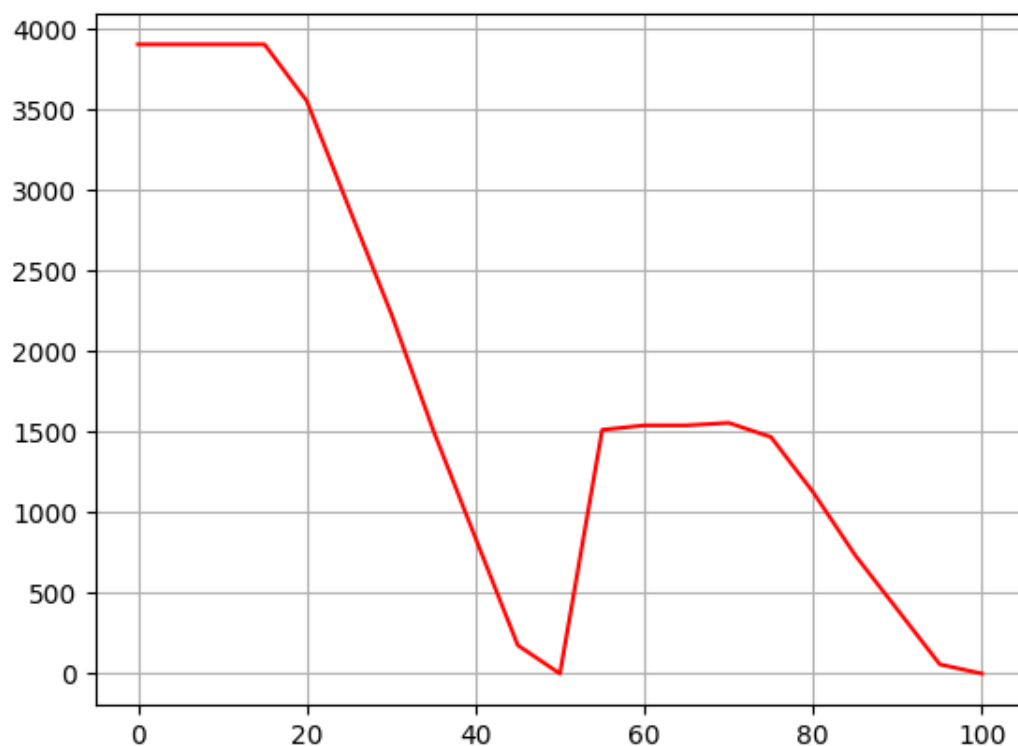
## Последние записи журнала

```
[ 132.539400] oom-kill:constraint=CONSTRAINT_NONE,nodemask=(null),cpuset=user.slice,mems_allowed=0,global_oom,task_memcg=/user.slice/user-1000.slice/user@1000.service/app.slice/app-org.gnome.Terminal.slice/vte-spawn-e5fc8112-d3c5-45a3-af4f-3d25fc53eee4.scope,task=mem2.sh,pid=2426,uid=1000
[ 132.539412] Out of memory: Killed process 2426 (mem2.sh) total-vm:3422684kB, anon-rss:1829248kB, file-rss:1792kB, shmem-rss:0kB, UID:1000 pgtables:6736kB oom_score_adj:200
```

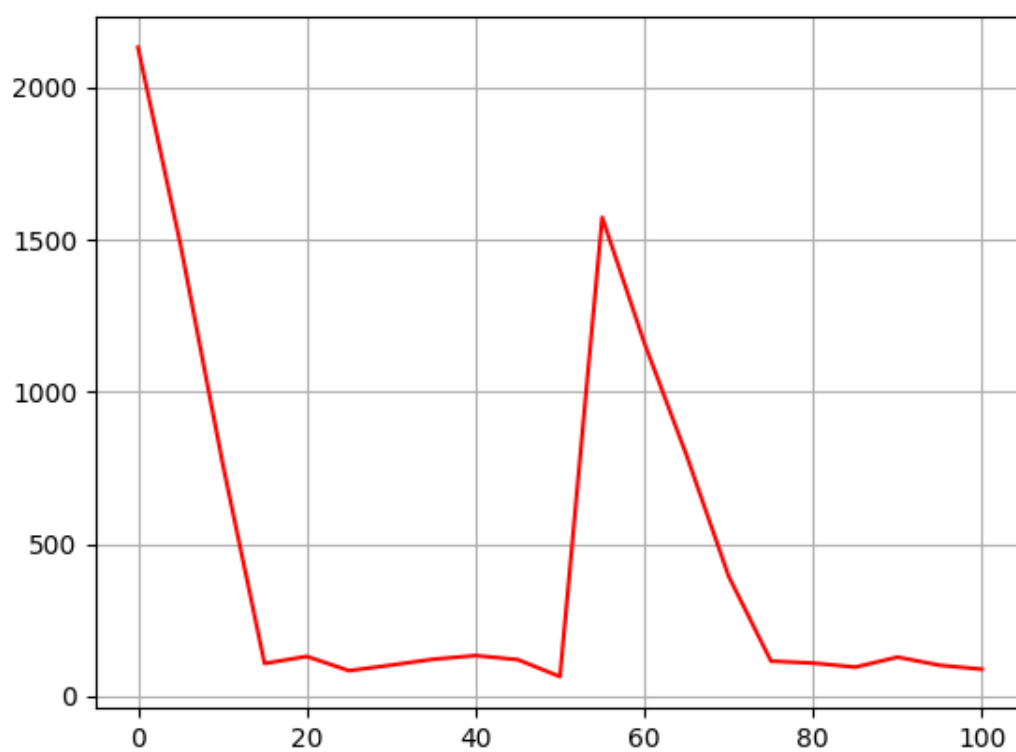
**Значение в последней строке файла report.log = 87000000**

**Значение в последней строке файла report2.log = 43000000**

**График зависимости swar (подкачки) от времени**



**График зависимости mem (свободной памяти) от времени**



**Изменения в верхних пяти процессах лежат в top.log. Лидирующие позиции занимают mem.sh и mem2.sh**

## **Вывод**

Когда скрипт mem2.bash был прерван и практически все доступное пространство памяти было выделено для скрипта mem.bash, все страницы памяти, зарезервированные для процесса mem.bash, были перемещены на раздел подкачки (swp). Впоследствии, при прерывании процесса, всё выделенное для него пространство памяти было освобождено.

## **Эксперимент №2**

При  $k = 10$ ,  $n = 88000000/10 = 8800000$ . Запуски завершились успешно, в системном журнале нет записей об аварийной остановке newmem.bash.

При  $k = 30$ ,  $n = 88000000$ . Запуск завершился аварийно, ubuntu перестал отвечать на запрос и требовал перезапуска, в системном журнале появились записи об 10 процессах. Это произошло, так как скрипты стали требовать памяти больше, чем есть у RAM.

Максимальное  $n = 6750000$  при  $k = 30$ , которое было найдено с помощью запуска разного количества скриптов(бинпоиска). Записей в системном журнале не появились, на этой потребовалось много времени, так как ubuntu падал и записывал результаты спустя какое-то время.