**Лабораторная работа №5**

Общий объем оперативной памяти: 1870900 кБ

Объем раздела подкачки: 839676 кБ

Размер страницы виртуальной памяти: 4096 байт

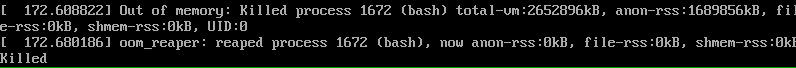
Объем свободной физической памяти в ненагруженной среде: 1713368 кБ

Объем свободного пространства в разделе подкачки в ненагруженной среде: 742820 кБ

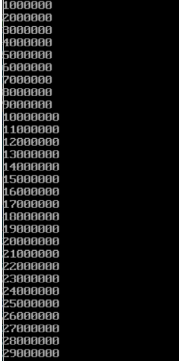
(посмотрел с помощью cat /proc/meminfo)

**Эксперимент №1**

Записи системного журнала



Записи в **report.log**



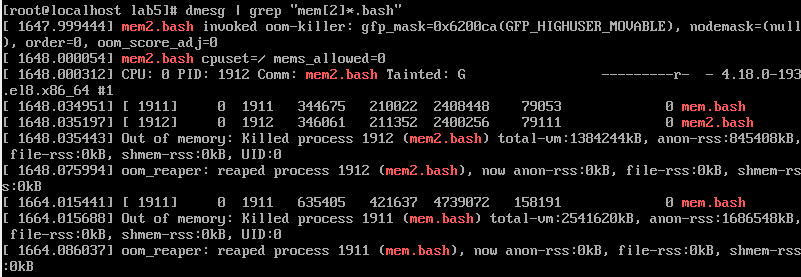
Изменение количества свободной памяти (в MiB) в процессе выполнения скрипта (шаг равен 3 секунды)

Процент потребления памяти процессом во время выполнения (шаг равен 3 секунды)

Во время начала использования процессом swap-памяти, в процессах появляется kswapd0 и не пропадает со второго места в top до аварийного завершения.

Второй этап

Записи в журнале:

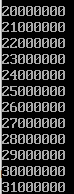


Записи в **report.log**

A picture containing text

Description automatically generated

Записи в **report2.log**



Изменение количества свободной памяти (в MiB) в процессе выполнения скриптов (шаг равен 3 секунды)

Процент потребления памяти процессами во время выполнения (шаг равен 3 секунды)

После начала использования swap-памяти, в top появляется процесс kswapd0, после аварийного завершения первого процесса он исчезает, потом когда начинает использоваться swap-память для второго процесса, он снова появляется до аварийного завершения второго процесса.

**Эксперимент №2**

Скрипт gen.bash получает аргументами K и N ($1 = K, $2 = N) и вызывает K раз newmem.bash с аргументом N.

При K = 10, N = 3100000 все запущенные скрипты успешно завершаются.

При K = 30, N = 3100000, сначала уменьшается свободная память, в момент, когда она начинает заканчиваться, начинают использоваться swap-память. В момент, когда количество свободного пространства подкачки опускается до примерно 100 MiB, скрипты начинают завершаться аварийно.

Подбор максимального N:

При N = 1000000 все скрипты отрабатывают, причем настолько быстро, что они даже не успевают появиться в top с частотой обновления 1 сек.

При N = 3100000 аварийная остановка произошла примерно спустя 25-30 секунд после запуска gen.bash.

При N = 2000000 все скрипты отработали без аварийных остановок, причем swap-память не была использована.

При N = 2600000 не было аварийных остановок, но на пике оставалось всего 200 MiB swap-памяти.

При N = 2700000 не было аварийных остановок, свободная swap-память снизилась до 80MiB и произошла аварийная остановка

Таким образом максимальное N, при котором все скрипты отрабатывают без аварийных остановок – примерно 2600000, что всего лишь на 500000 меньше, чем 1/10 от максимального N, полученного в первом этапе эксперимента 1. Другими словами, засчет распределения потребления памяти на большое количество процессов (запускаемых с небольшим интервалом), мы можем использовать значительно большее количество памяти, засчет оптимального распределения системой swap-памяти.