CS401 - ASSIGNMENT #2

Estimation of RSS (Resident Set Size)

What is RSS?

RSS or resident set size is the amount of actual physical memory that is used by the process. Only a subset of actual virtual memory allotted to the process will be in physical memory at any given point in time. It is called RSS.

How is RSS calculated?

Kernel manages memory in units of Pages. Pages are small sized chunks of bytes. It is usually of size 4KB in a 32 bit system. Page table keeps track of all the pages allotted to a process. Each process has its own page table.

The procedure I used to calculate RSS is to walk through the page table entries and check if the page is present in memory or not. Page table had Page Table Entries (PTE). The PTE contains many bits which indicate the status of the page. The bit that is of importance here is the Valid bit / Present Bit. If valid bit is set, then we can count that page is available in physical memory.

The kernel module iterates over all PTEs of a process and checks if the page is present in physical memory and counts if it is.

High Level Algorithm

1. Get task_struct of process we want to calculate RSS
2. For each VMA region inside mm_struct
    2.1. For each address incremented in PAGE_SIZE units
        2.1.1. Get Page Table Entry
        2.1.2. If Present Bit is set, add to RSS_SUM
        2.1.3. Repeat until end of VMA region
    2.2. Repeat until end of VMAs
3. Print RSS_SUM

How to verify Page Size?

It can be verified using /proc/meminfo and /proc/vmstat.

PAGE_SIZE = Mapped/nr_mapped

How Page Table is organized?

Page Tables in linux are organized as 3 levels. They are

1. Page Global Directory (PGD)
2. Page Upper Directory (PUD)
3. Page Middle Directory (PMD)

To access a page table entry (PTE), we need to go through these levels.
The base pointer of page table of a process (ie PGD) can be found in mm_struct inside task_struct. (`mm_struct→pgd`). The kernel provided methods `pgd_offset(),pud_offset(),pmd_offset()` and `pte_offset()` provides access to PTE. The present bit can be checked using `_PAGE_PRESENT` flag.

How to compile the module?

1. Change directory to ./rss/modules/
2. `make`
3. The file rss.ko will be generated and it is the module file

How to execute ?

1. Go to folder where rss.ko resides
2. Find the pid of the process which you want to find the RSS using the command "ps -e"
3. `sudo insmod rss.ko pid=PID ntimes=N` # where PID is you process id and N is the number of times rss should be calculated in 10 second intervals

   Eg: `sudo insmod rss.ko pid=4825 ntimes=4`

4. After insertion, you can unload the module using
   `sudo rmmod rss`

How to get results?

Results are printed in system log. You can access it using the command
`dmesg`

How to verify results?

There are several ways to see the RSS of process

1. `ps -eo pid,comm,rss`
2. `cat /proc/<PID>/status`
3. `cat /proc/<PID>/smaps` #(detailed)

Sample OUTPUTs

[1919.366486] Page size : 4096
[1919.366492] Process name with PID 3976 is : leafpad
[1919.123180] RSS : 19448 VMA : 180824
[1929.343980] RSS : 19450 VMA : 180974
[1939.393080] RSS : 19508 VMA : 181014
[1919.385838] Module exit

[1699.017901] Page size : 4096
[1699.017907] Process name with PID 3218 is : firefox
[1699.041451] RSS : 150908 VMA : 653908
[1699.048878] Module exit

[9323.979660] Page size : 4096
[9323.979666] Process name with PID 4643 is : abiword
[9323.985226] RSS : 69480 VMA : 299520
[9334.016119] RSS : 69492 VMA : 299520
[9344.032049] RSS : 69492 VMA : 299520

[9354.048056] RSS : 69492 VMA : 299520
[9362.568673] Module exit

[9385.723683] Page size : 4096
[9385.723688] Process name with PID 1 is : init
[9385.724002] RSS : 3816 VMA : 4708
[9395.743463] RSS : 3816 VMA : 4708
[9405.759497] RSS : 3816 VMA : 4708
[9415.775516] RSS : 3816 VMA : 4708
[9433.777809] Module exit

[2021.508745] Page size : 4096
[2021.508750] Process name with PID 3140 is : sublime_text
[2021.518569] RSS : 52044 VMA : 516028
[2031.520083] RSS : 52044 VMA : 516028
[2041.571925] RSS : 50324 VMA : 532412
[2064.086957] Module exit