Theory Assignment 1
CO21BTECH11004
DARPAN GAUR

Que 1) Describe a mechanism for enforcing memory protection in order to prevent a program from modifying the memory associated with other programs.

a) Do you see any other mechanism for enforcing the above requirement?

Solution)

Protection of memory using dynamic tainting:- Dynamic tainting is a technique for protecting programs from illegal memory accesses. When memory is allocated at runtime, this technique taints both the memory and the corresponding pointer using the same taint mark. Taint marks are then suitably propagated while the program executes and are checked every time a memory address *m* is accessed through a pointer *p*; if the taint marks associated with *m* and *p* differ, the execution is stopped, and the illegal access is reported.
Source:- Wikipedia

b) While memory protection is necessary, there are several examples where programs require memory to be shared between them. Please give two such examples.

Solution)

Examples where programs require memory to be shared between them are as follows:-

- Every executable on Linux requires libc.so library can execute the program executable. So while executing many processes simultaneously where each process requires this library libc.so, OS loads libc.so to a specific virtual address space range of every process mapped to a fixed physical address range in main memory. So every process will refer to that fixed physical address range to execute any code in libc.so. So, in this case,, every process also shares some physical address ranges. Here these processes are reading usings shared memory. If OS allocate separate memory for each process for accessing libc.so library, then if processes are in high number then we can imagine the level of duplication of libc.so library. So, shared memory helps alot in this.

- Server and Client system:- A server process creates file containing data, maps it, and initializes it to a value. After that, a client process opens the shared file, again maps it, and checks that the data is correctly initialized. Here communication is done via shared memory where changes made by one process can be viewed by another process. Eg:- Client and database server uses shared memory for faster access. Any client can connect over shared memory to any database server that runs as a service. Shared memory provides fast access to a database server, but it poses some security risks.

Source:- Wikipedia

Que 2) Please study the PCB (Process Control Block) structure of a Linux system from any reliable Internet source. Describe any 10 fields in the PCB of a process in the latest Linux Operating System. Please cite your source as well, like in question 2.

Solution) Each process is represented in the operating system by a process control block (PCB)—also called a task control block.
Fields in the PCB of a process in the Linux Operating System are as follows:-

- **Process state:-** The state may be new, ready, running, waiting, halted, and so on.
- **Program counter:-** The counter indicates the address of the next instruction to be executed for this process.
- **CPU registers:-** The registers vary in number and type, depending on the computer architecture. They include accumulators, index registers, stack pointers, general-purpose registers, and any condition-code information. Along with the program counter, this state information must be saved when an interrupt occurs to allow the process to be continued correctly afterward when it is rescheduled to run.

- **CPU-scheduling information:** This includes a process priority, pointers to scheduling queues, and other scheduling parameters.
- **Memory-management information:-** This information may include such items as the value of the base and limit registers and the page tables, or the segment tables, depending on the memory system used by the operating system.
- **Accounting information:-** This information includes the amount of CPU and real-time used, time limits, account numbers, job or process numbers, and so on.
- **I/O status information:** This includes the list of I/O devices allocated to the process, a list of open files, and so on.
- **List of Open Files:-** It contains the information of all the files required by the program during its execution. This information is also helpful for the operating system because it helps the system close all opened files which are not closed explicitly at the program's termination.
- **Process-id (PID):-** When a process is created, a unique id is assigned to the process used for unique identification of the process in the system.
- **PCB Pointer:-** In this field, the pointer has an address of the next PCB whose process state is ready. In this way, the operating system maintains the hierarchy of all the processes so that a parent process can locate all the child processes it creates easily.

Sources:-

- Operating System Concepts *Tenth Edition,* OS-BOOK
- https://binaryterms.com/process-control-block-pcb.html
- Wikipedeia