Byron Laferriere

Professor Hardikar

IT-365

Module 5 ‘Run The Code’ Questions

***Challenge Question One:*** *On a system with paging, a process cannot access memory that it does not own. Why? How could the operating system allow access to other memory? Why should it or should it not?*

Processes use an address that is translated through its page table and therefore, cannot access the memory allocated to other processes. The operating system can permit access to a process to access other memory by establishing physical address entries in the system to be accessed by processes looking to communicate or by offering system calls to read/write from other processes currently running. I think that this should be done by operating systems because it seemingly creates better process communication that is fast and cheaper than some other options.

***Challenge Question Two:*** *What is the purpose of paging the page tables?*

Paging is a practice that is used when page tables become too large for the system they are operating on. It is a memory-management scheme that allows the physical address space of process to be contiguous (Silberschatz et al, 2009). When physical address space or RAM is overwhelmed by a page table, the system mapping is heavily impacted and slows down. Paging is used for this purpose and redistributes the page table so that outer pages hold references to inner page tables. By requiring internal fragmentation of frames and not external, it allows for better varying page size translations. This makes the inner page table suitable for the size of the physical address space or RAM. With the way that pages have grown in size recently, due to processes, data sets, and main memory becoming larger, this is preferrable for operating systems (Silberschatz et al, 2009).

Reference:

Silberschatz, A., Galvin, P. B., & Gagne, G. (2009). *Operating System Concepts with Java* (8th ed.). John Wiley & Sons.