CSCE 410/611 Operating Systems Spring 2023

Homework for Week 5

(Due Date: Check Canvas)

1. A colleague suggests a trick to reduce the pressure on the swap device. Instead of swapping out pages that belong to executable code into the swap area, the operating system could just not swap them out and just take away the memory frames. If the code page is needed later, it could be paged back in directly from the file containing the executable. Her argument is that the OS will save time by not storing the code page into the swap device, and will save space on the swap device. The executable code exists on disk anyway, and could be fetched from there. Would you approve of this design? Why or why not?

I would not approve of this design at its current state. What happens if the page of the main executable is swapped out and modified then swapped back in. Then the current program is running an old binary where the references and addresses are modified. This in the best case would cause a crash and reboot and the worst case some memory modification. For this to be approved, a lock system would need to be implemented preventing a user from changing the file on the disk.

References

[1] A. Silberschatz, P. Galvin, and G. Gagne, *Applied Operating Systems Concepts*, John Wiley & Sons, Inc., New York, NY, 2000.

[2] Deitel, Deitel, and Choffnes, *Operating Systems*, Pearson / Prentice Hall, 2004. [3] A. S. Tanenbaum, *Modern Operating Systems*, Pearson / Prentice Hall, 2008. [4] L. F. Bic, A. C. Shaw, *Operating Systems Principles*, Prentice Hall 2003. [5] C. Crowley, *Operating Systems, A Design-Oriented Approach*, Irwin 1997. [6] M. Herlihy, N. Shavit, *The Art of Multiprocessor Programming*, Elsevier, 2008

1