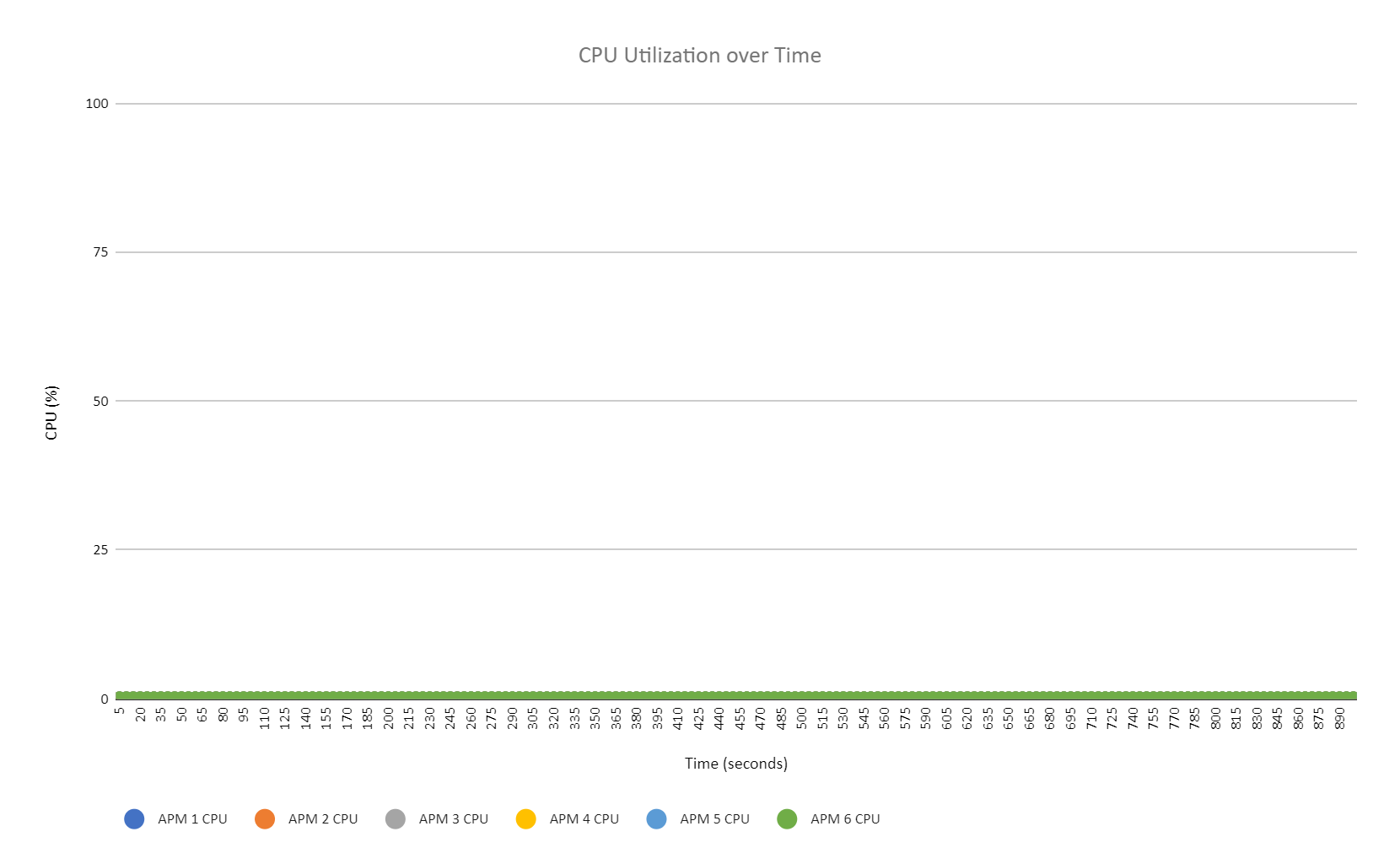
**NSSA-220 Project 1: Application Performance Monitoring**

**Luke Chrampanis, Dylan Shanely, Mikos Panagakos**

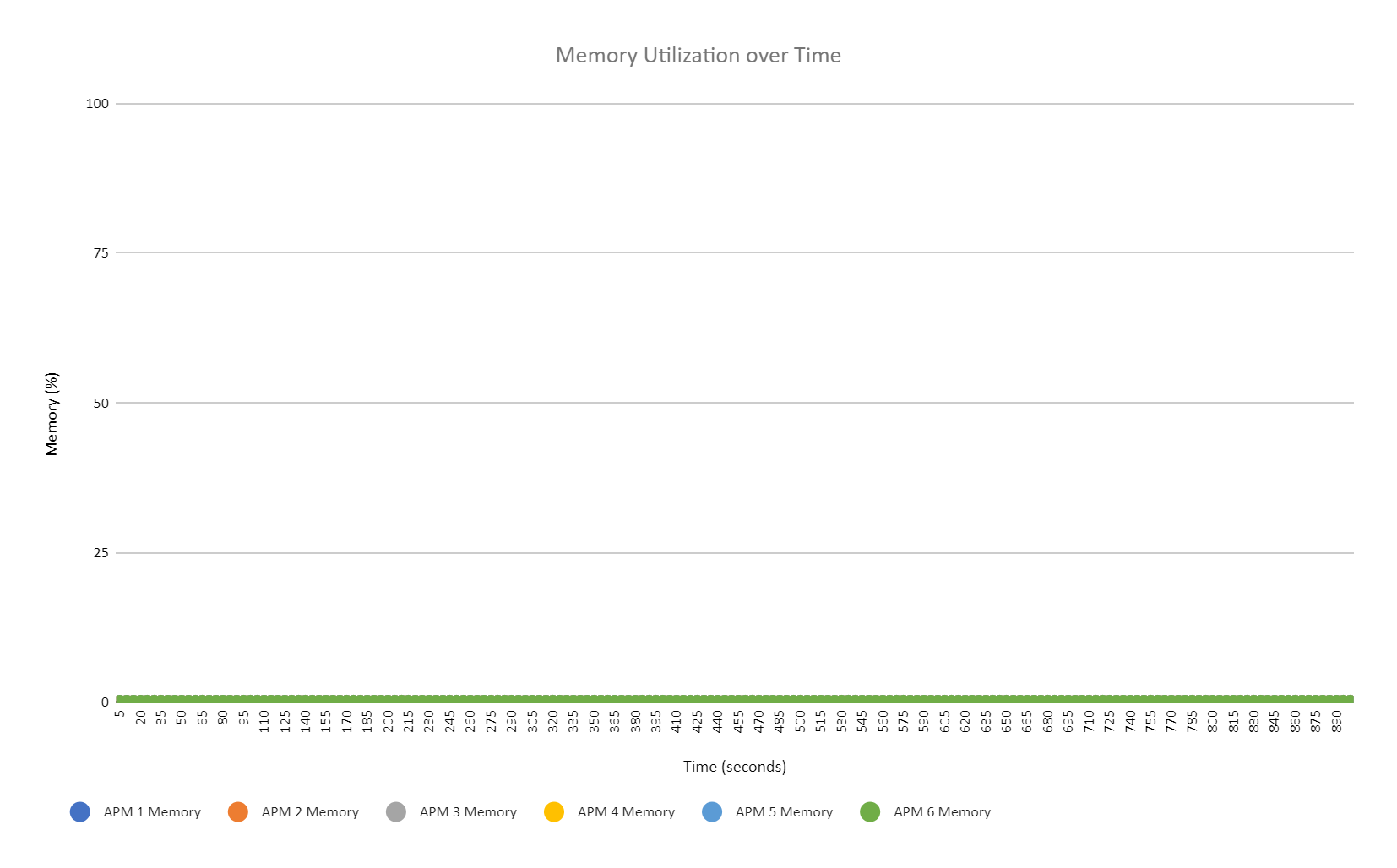
**Introduction**

This project was about collecting Application Performance metrics by monitoring the performance of running 6 sub processes connecting to other machines. This project involved writing a script to run 6 provided applications and collect the process and system level metrics as a result of running these 6 applications. The goal of this project was to interact with the operating system in bash and learn skill involving monitoring a machine in order to practice security monitoring and learn applicable bash skills.

**Process Level Metrics**

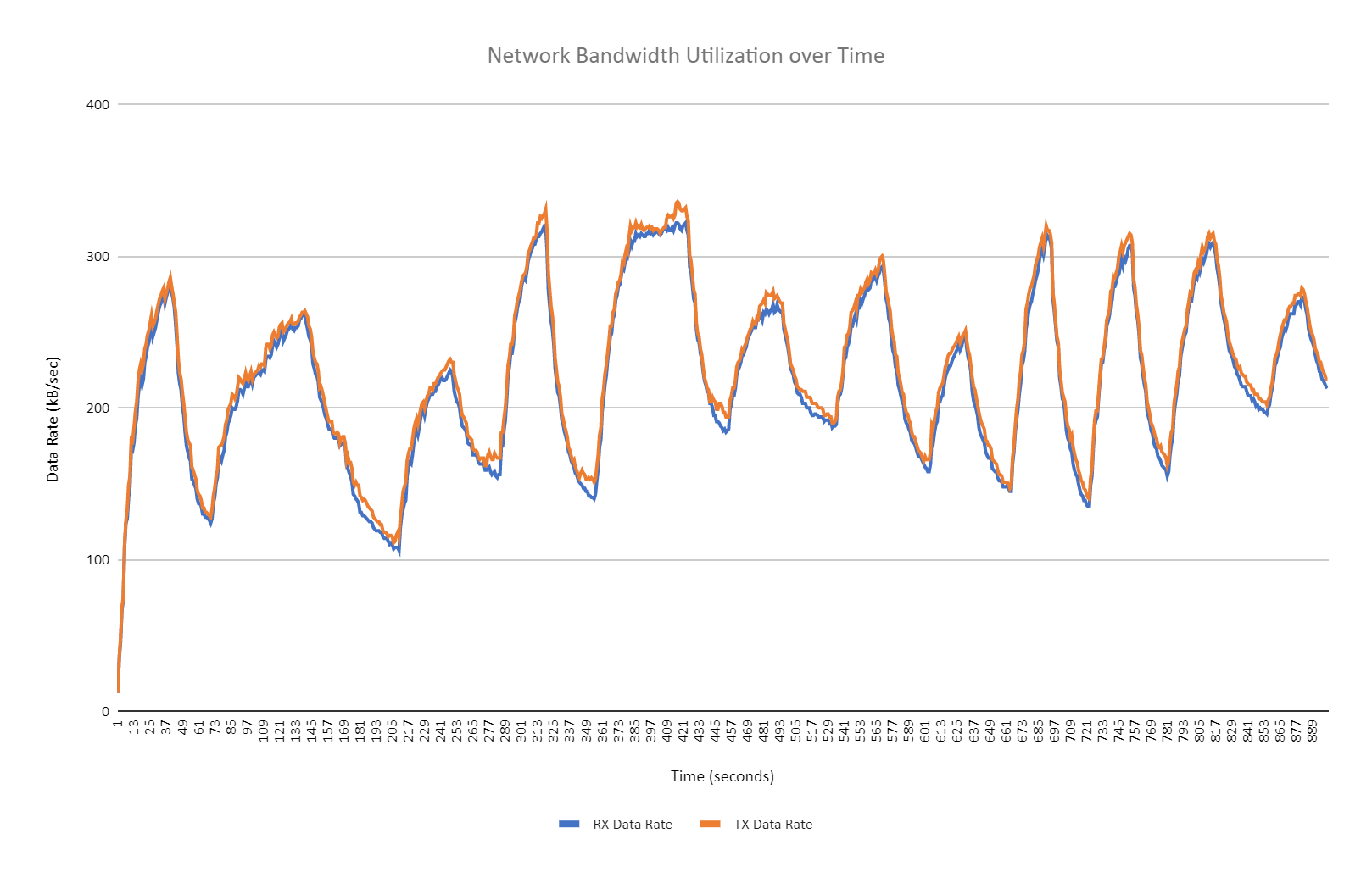


The CPU Utilization over Time plot shows that the CPU was never utilized in any of the 6 APM processes. This suggests that the processes were not complex enough to strain the CPU into utilization.

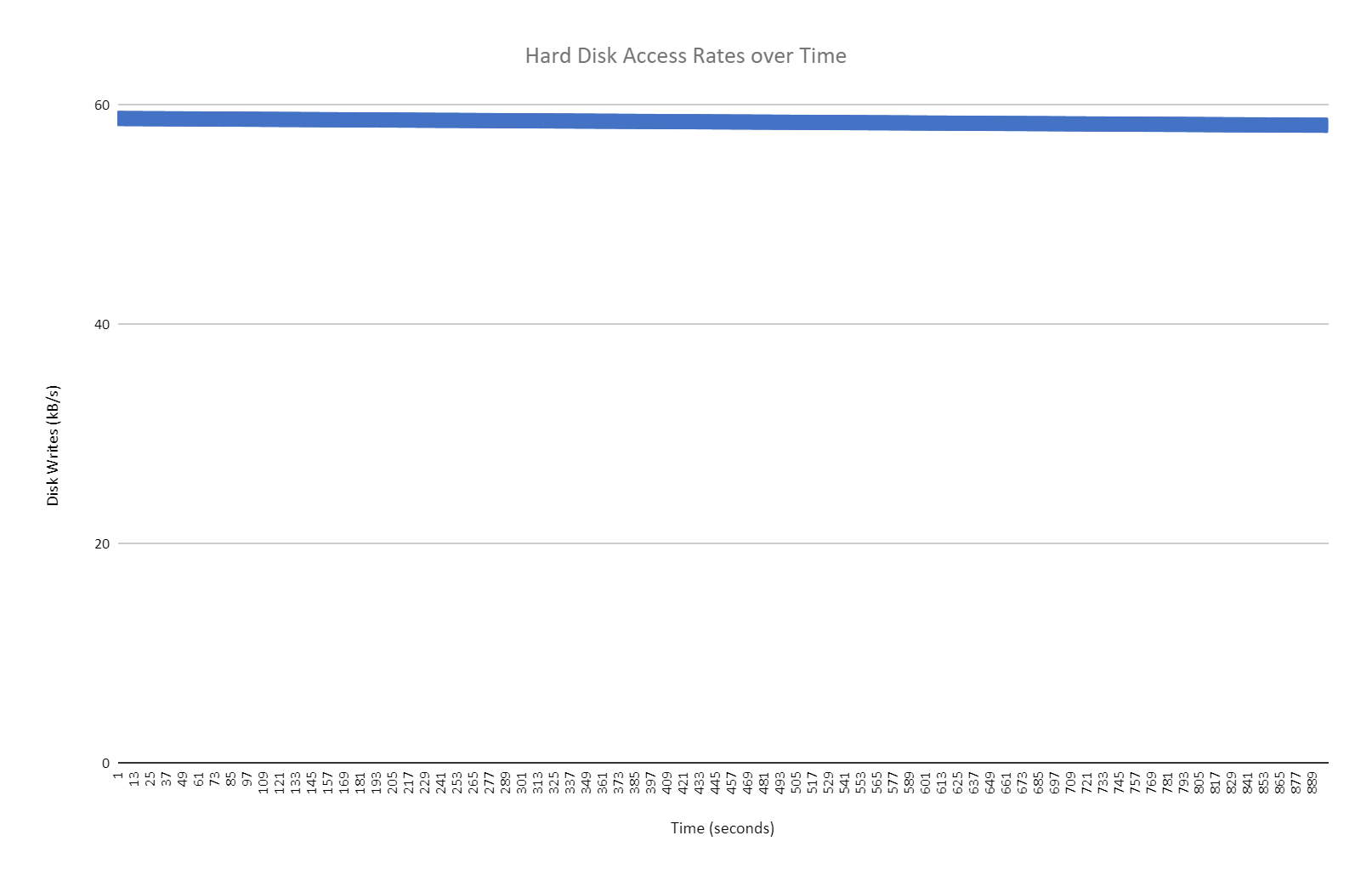


The Memory Utilization over Time plot shows that the memory was never utilized in any of the 6 APM processes. This suggests that the processes were not complex enough to strain the memory into utilization.

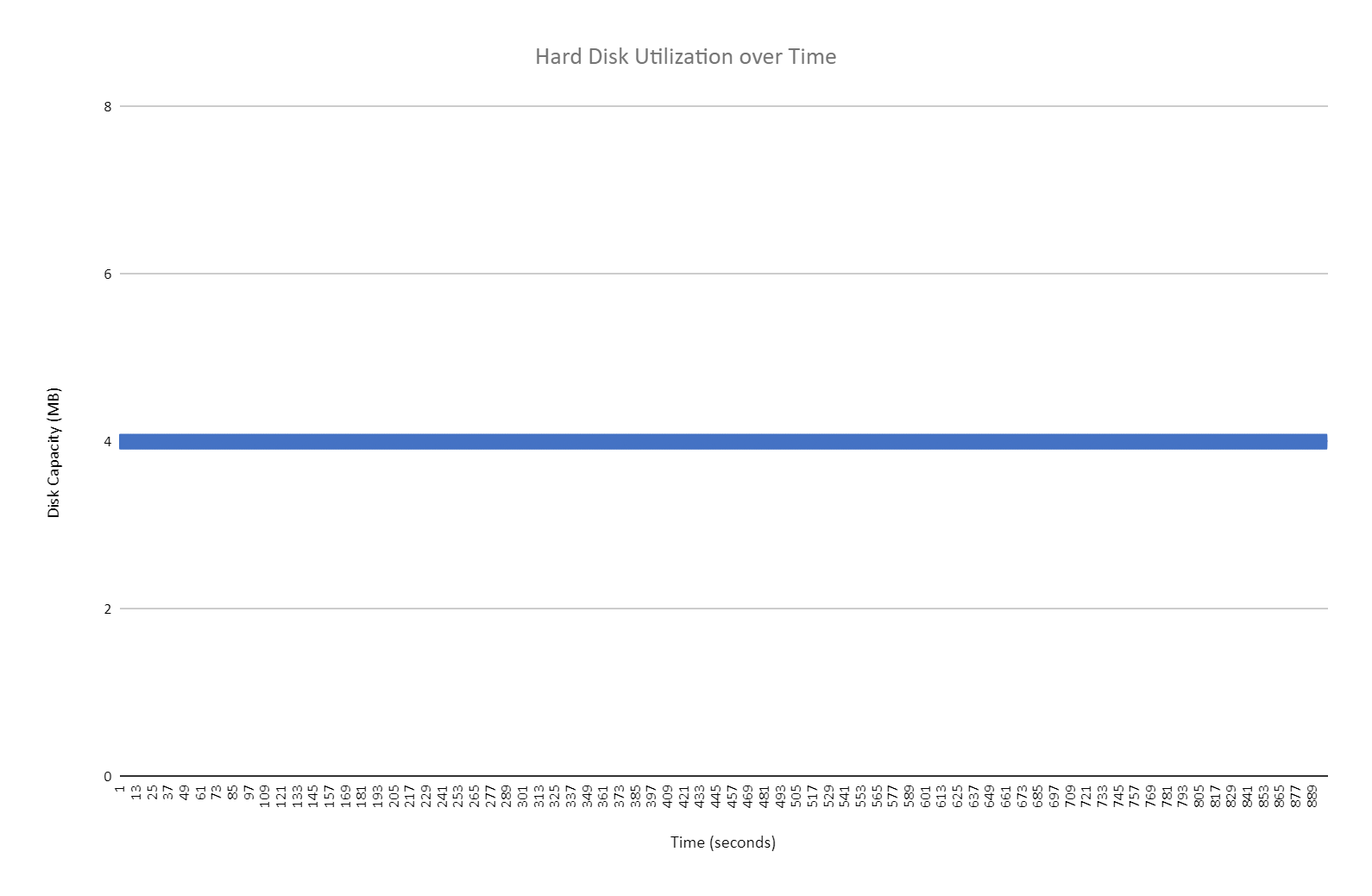
**System Level Metrics**



The Network Bandwidth Utilization over Time plot shows that receive and transmission data rates stay very close to each other and rise and fall on similar intervals of time. There was a slow overall increase in rates as the data rates rose and fell suggesting that the network slowly was better hosting the data transmission of the computer.



The Hard Disk Access Rates over Time plot shows a constant disk access rate of ~58kb/s. This suggests that the disk’s access rate was not being affected over time and overall remained stagnant for the duration of the test.



The Hard Disk Utilization over Time plot shots a constant disk capacity of 4MB. This suggests that the data being written by the test over 15 minutes did very little to affect the capacity of the disk.

**Summary and Lessons Learned**

The VM had more than enough resources to run these 6 processes. The processes were very minimal and failed to stimulate process level metrics as gathered by the ps -aux command. However it did stimulate system level metrics, suggesting that the processes were correctly running and data was being correctly collected. If anything, the VM had too many resources to valuably display strain and achieve the result the project was intended to display. Regardless, our team still learned valuable skills in bash including but not limited to, interaction with process and system level os commands, multiprocessing, and team-based coding skills.