**Introduction**

This project consisted of creating an application performance monitoring scripting tool using the command language of bash. The purpose of this tool is to measure resource utilization in the operating system known as Linux, while providing the user with metrics collected on the process and system levels. Through the utilization of this tool, the client will be able to detect systematic/ malicious issues in software, and determine if the computing resources provided can handle the mix of software in use.

**Process Level Metrics**

This CPU utilization plot specifies how much of the processor’s capacity is currently in use by the system. The third process we ran used the most CPU, with 64.5% being in use shortly after a minute of it running. The first and second procedures used the least, with 0% of the CPU being utilized during the 15-minute run period. After further analyzing the graph, we noticed that APM4’s CPU usage gradually increased over time, while APM3 and 5 started off with using a high percentage of the CPU, but later stabilizes itself.

This memory utilization plot indicates the amount of inside memory (RAM) currently in use by the system. The fifth process utilized the most memory, with 24.1% being its peak as it followed a non-monotonic pattern. The third process utilized the least, as none of the memory was in use as it ran. After examining the chart, we discovered that there were two memory leaks in the running of the sixth process.

**System Level Metrics**

The network bandwidth utilization plot signifies the amount of bandwidth each network interface card is using over time. As one can evaluate from the graph, the receive data rate is hardly visible due to the transmit data rates being nearly identical. This is a bimodal distribution graph with two main peaks at certain points.

The hard disk access rate plot implies disk access activity over time. Although there are no noticeable patterns, we can see that the graph gradually increases before plateauing off towards the end of the run of the program.

This hard disk utilization plot monitors the remaining disk capacity. The data formed a step chart which can be useful to display changes that occur at irregular intervals.

**Summary**

Overall, after running our code and collecting the data, we concluded that our virtual machine (VM) did in fact have enough resources to handle the mix of application processes that were running. Throughout the multiple tests runs that we conducted, the VM did not crash at all, and it handled the workload appropriately. No problems were detected regarding the machine slowing down, and all the tests ran smoothly.

**Lessons Learned**

Through this assignment, we learned about the significance and practicality of an application monitoring tool and how it can be applicable in the real world. The data we collected through each process gave us an insight as to how system administrators can use that information to troubleshoot any issues or make updates/ upgrades if necessary. Subsequently, we learned about the importance of monitoring metrics on a process and system level which can be extremely useful to know in industry. Consequently, we had many takeaways from this projects and in addition to learning how to work with new coding commands, we also learned new terminology along the way.