An Exploration Scrutiny of Systems Performance Monitoring Tools:

An Applications and Agentless Server Approach

Phumzile Nomnga [1] and Mfundo Shakes Scott

[1] Department of Computer Science, University of Fort Hare, P/Bag X1314, Alice, 5700, RSA
Tel; +27 (0) 40 602 2745, Fax: +27 (0) 86 248 9404

[2] Department of Computer Science, University of Fort Hare, P/Bag X1314, Alice, 5700, RSA
Tel; +27 (0) 40 602 2745, Fax: +27 (0) 86 248 9404

Abstract

Performance monitoring tools play a critical role in the overall productivity of business organizations by providing the necessary means of measuring the performance of the system or server attributes of the organization. They measure system or server throughput and processor usage through programmed software or technological platforms. Their monitoring is extremely efficient such that Network administrators normally rely on them wholeheartedly and without any doubts. Their monitoring statistical results normally compel the Network Administrators to either believe that the system or server performance is normal or else make decisive interventions on system or server performance. The efficiency of these performance monitoring tools builds confidence and reliability in Network Administrators such that they always view the statistical results as accurate and genuine. The responsibility of measuring the performance accuracy of the performance monitoring tools normally does not constitute the duties and responsibilities of Network Administrators. This, therefore, result into a situation whereby this essential responsibility is not executed under the presumption that the performance monitoring tools are accurate and genuine. This holds in various business organizations despite the fact that some of these performance monitoring tools may poorly perform under certain server or system conditions. It is with this background that this research project is undertaken, which is to conduct an exploration scrutiny of applications and Agentless server performance monitoring tools. The basis behind the exploration scrutiny is to advise and provide expert guidance to Network Server and Applications Administrators on what to monitor, why to monitor, and optimal threshold.

***Keywords* —Agentless Server, Application Performance Monitoring Tools, Scrutiny, Heroix Longitude, OpManager and ManageEngine**

I. INTRODUCTION

This research study is undertaken to conduct an explorative scrutiny of Applications and Agentless Server Performance Monitoring Tools. We conducted an exploration scrutiny on the performance monitoring tools with the understanding of their critical role in the general network. “In today's world of networks, it is not enough simply to have a network; assuring its optimal performance is Key” [1]. To achieve this purpose, the ManageEngine OpManager and ManageEngine Application Manager, as Agentless server and application performance monitoring tool, has been used as case studies. However, various other performance monitoring tools such as Heroix Longitude and AppPerfect Agentless monitor have also been brought forth into the investigation to yield an informed and objective scrutinization analysis. The throughput of each and every performance monitoring tool under investigation was compared and analyzed with respect to the server baseline prior being monitored and analyzed in comparison with other performance monitoring tools. Any of the investigated performance monitoring tools that produced known and unknown results that are equal or similar to the server baseline has been recommended from the results of this research project. It was envisaged that the recommendations be vividly specific about the platform(s) to which the performance monitoring tool(s) can be deployed.

II. RELATED WORK

Computer technology has gradually placed itself in the centre of human development ever since its inception. As a direct consequence of this, several research studies have been conducted to maintain and advance the standard of technology around the world. A research study was conducted on “Statistical Analysis System (SAS) Application Performance Monitoring for Unix” [2] and the research focused on various aspects of performance monitoring in SAS environment. The research study used, amongst other things, the importance of performance baseline as it regards to the SAS application environment. In establishing the performance baseline, the research study captured the performance data over a period of time and multiple periods were observed. This aided the research study to compare the performance baseline with the current data to easily detect differences that may be caused by a change in server performance.

Furthermore, a research on “Optimum Network Performance (OPNET) Modeler and Ns-2: Comparing the Accuracy of Network Simulators for Packet-Level Analysis using Network Testbed” [2] was also conducted to provide guidance to researchers employing packet-level network simulations. This research was conducted as a comparison of the aforementioned simulators to a live network testbed output. During the experimental comparison, both Calculator –Based Ranger (CBR) data traffic and a File Transfer Protocol (FTP) session were deployed on the simulators and network testbed. The research study revealed that there is a "necessity of fine-tuning the parameters within a simulator so that it closely tracks the behavior of a real network."[2]

Lastly, a "Qualitative Comparison of Network Simulation Tools" [3] was conducted as a research study to provide a decisive argument on which simulator should be used. OPNET Modeler and Ns-2 were also used as case studies to this research work of Qualitative Comparison of Network Simulation Tools. The research study argued that "since the results of simulators should be as significant as possible, a high quality of simulator is indispensable" [3].

It is meet to highlight that there are a few papers and studies that can be found making an account on Agentless server and application performance monitoring tools and this worsened when it came to their comparison, exploration and scrutiny.

III. RESEARCH OBJECTIVES

Drawing from our own experiences in the field of Information and Communication Technology (ICT), we have seen and learned that the Agentless performance monitoring tools used in the ICT field to monitor network and infrastructure are taken for granted as perfect, relevant, accurate and reliable tools. Network Administrators tend to be oblivious to the fact that "Agentless data collection is a powerful technology which has its advantages and disadvantages" [4]. Some of the Network Administrators do not even comprehend and subscribe to the notion that "Before software is put into operation phase, in order to check software for bugs, it must be thoroughly tested" [5]. Agentless performance monitoring tools are assumed by Network Administrators to be perfect and accurate and are used as such. This misconception compels Network Administrators to lightly consider that "it is a common phenomenon that software performance and quality of service (QoS) degrade over time" [6]. This "calls for continuous monitoring of applications in order to determine whether QoS is kept on a satisfactory level [4]". On the basis of these critical circumstances, we hereby coined our research objectives as follows:

- **Objective 1: Investigate the accuracy of the Agentless performance monitoring tools**
- **Objective 2: Answer the question – shall we rely on the statistical results which are presented by Agentless performance monitoring tools?**
- **Objective 3: Compare and contrast the performance of the performance monitoring tools**
- **Objective 4: Offer advice and provide expert guidance on server and application monitoring using Agentless performance monitoring tools**

The primary step towards achieving the aforementioned objectives is to possess accurate knowledge about every single performance monitoring tool.

IV. RESEARCH METHODS

This section relates to the methods that were used to conduct a comparative analysis of various Agentless server and application performance monitoring tools. It depicts how the research was constructed from the research title, data collection, and procedure for data analysis and comparison.

1. Research Construction

The foundation of this research was literature review which has assisted in broadening the understanding of what is known and unknown regarding the performance monitoring tools. Literature review in essence is a "systematic search to find out what is already known about the intended research topic" [7]. This required an extensive reading of publications, research papers, textbooks and articles. Not all reading material read was of interest to this research; however, it was the only process to follow in order to find the relevant reading material to this research.

2. Data Collection

The search engines such as Google, scholar.google have a tendency of producing outcomes that are not really of interest which might lead the researcher to deviate from the research point of interest. To guard against this, certain key words had to be used during online searching of relevant reading material. To find the related works to this research, key words such as "baseline, application performance monitoring tools, technical comprehension, comparative analysis, OpManager, ManageEngine, Up.Time, Heroix Longitude, SolarWinds and Agentless server", were used during data collection. Furthermore, trial versions of OpManager, ManageEngine, SolarWinds, Up.Time and Heroix Longitude were downloaded and deployed to the VMware ESXi virtual server of the University of Fort Hare, Computer Science Department laboratories. The virtual server was accessed through VMware vSphere Client and was also used as a platform to deploy our performance monitoring tools. It is through the User Interfaces (UI) of the vSphere Client that the performance monitoring tools were explored. And the performance output they individually generated was used to conduct a comparative analysis of their monitoring.

3. Procedure for Data Analysis

The data obtained through the deployment of OpManager, ManageEngine Application Monitor, SolarWinds, AppPerfect and Heroix Longitude was then analyzed with respect to the objectives of this research. The following questions were then posed:

- What is the amount of data collected about each Key Performance Indicator (KPI)?
- Is there any difference on the amount of data collected by different performance monitoring tools?
- If, yes how much difference there exists? and
- What is the difference compared to the performance monitoring tools?

This is the procedure that the research project adhered to in order to achieve the research objectives which are to explore and scrutinize our performance monitoring tools.

V. SYSTEM ARCHITECURE

The system architecture of this research work takes full advantage of the fact that "the Internet architecture has proven its worth by the vast array of applications it currently supports and the wide variety of network technologies over which it currently runs" [8]. In addition, the research project also took advantage of the virtualization so as to "improve system security, reliability and availability, reduce costs and provide greater flexibility" [9]. Figure 1, below, depicts the logical system architecture of the research project and is the system architecture of this research project portraying a typical network system where performance monitoring was conducted.

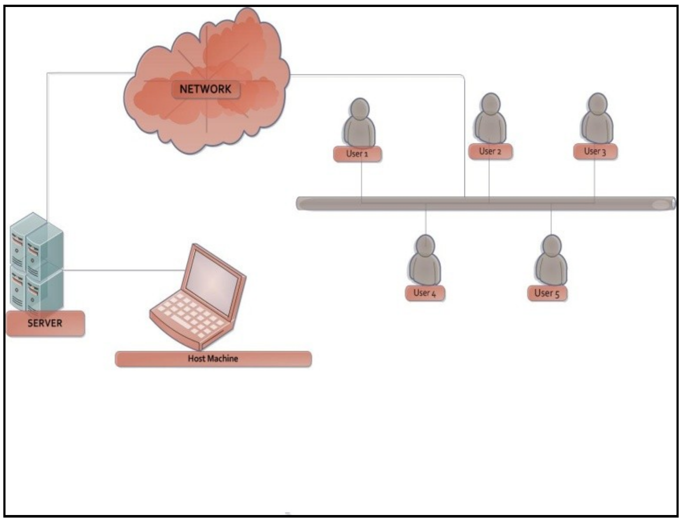

Figure 1: System Architecture

In figure 1 above, Users 1, 2, 3, 4 and 5 are all connected to the network in order for them to interact with the server. They run many different applications. Similarly to how active directory operates, wherein users will log into the system using their confidential user credentials and thus interact with the system in many different ways. Users 1, 2, 3, 4, and 5 provide a conducive platform for authentic users to log into the system and interact with it in many various ways as they like. Through the network, the applications initiated by the users are sent to the server where they are considered for further processing. This logical system architecture informed and guided our physical system architecture.

It is worth noting that through software's such as VMware vSphere Client "which provides health monitoring data for Elastic Sky X (ESX) hardware to support datacenter virtualization" [10], the monitoring process was of virtual form. On a single physical computer (Host Machine) in Figure 1, multiple independent virtual operating systems ran and were monitored. Assuming the role of the network administrator, we deployed different performance monitoring tools such as ManageEngine OpManager and ManageEngine Application Manager sitting on the Host Machine to the network and monitored their performance. Therefore this proves that the network administrator can monitor the system or server performance without physically going to the server location or necessarily having to make any physical contact with it. In this way, the research project has seen to the deployment of different performance monitoring tools, which were later analyzed and compared, into the virtual severs through the Host Machine.

VI. MONITORING FRAMEWORK

The Agentless server performance monitoring tools, namely, ManageEngine OpManager, Heroix Longitude and ManageEngine Application Manager, were all deployed virtually into the same platform i.e. University of Fort Hare, Computer Science Department Walk-in-Lab server and at the same time. Although the number of the KPIs that they are developed to monitor varies; on the servers perspective these three performance monitoring tools at least were characterized by the following common KPIs:

- **CPU utilization**
- **Disk utilization**
- **Memory Utilization**

Whereas on the applications perspective; ManageEngine OpManager and ManageEngine Application Manager, when monitoring the Postgre Structured Query Language (PQL) 172.20.56.72_PGSQL, were characterized by the following:

- **Availability**
- **Health**
- **Today's Uptime**
- **Today's Availability**

It is through the graphical presentations and statistical outcomes they provided relating to the system resources mentioned above that a critical exploration scrutiny was conducted between them. The usual suspects when identifying server performance degradation are CPU, Memory, and the Disk utilization [11] hence why this research project critically looked at them.

VII. PERFROMANCE STUDY

In this section we hereby present the results of the ManageEngine OpManager, Application Manager and Heroix Longitude monitoring with respect to the CPU utilization of the Walk-in Lab server. We sequentially commence accounting about OpManager, Application Manager and Heroix Longitude.

1) *CPU Utilization*

In this section we hereby present the results of the ManageEngine OpManager, ManageEngine Application Manager and Heroix Longitude monitoring with respect to the CPU utilization of the Walk-in Lab server. We commence our presentation with OpManager, followed by Application Manager and finally make an account on Heroix Longitude monitoring.

1) *Results: CPU utilization*

Figure 2, to follow, depicts the measurement and monitoring of the CPU utilization by OpManager and further presents the statistical results of such a process. The y-axis of the graph represents the amount of CPU utilization in percentages whereas the x-axis represent the time interval at which the CPU utilization was measured and monitored by OpManager. The time interval of measuring and monitoring the CPU utilization by OpManager was ranging from 00:00am to 22:00pm whilst the percentage sufficed to range from 0% to 100%. What had been apparent was that the server under investigation consisted of a multi-core processor hence why the figure 2 consists of two graphs. OpManager termed these processors CPU0 and CPU1 for clarity purposes because "OpManager supports a real time CPU utilization monitoring" [12]. To present the CPU utilization of Walk-in Lab server results, we made two distinct accounts of the CPUs, beginning with CPU0 and conclude by CPU1.

In the figure below it is presented that the minimum amount of CPU0 utilization was 0.0%, maximum being 68.0% and the average was 4.45% within the aforementioned time interval. Furthermore, we can learn from figure 2 that the minimum CPU0 utilization was measured and monitored by OpManager at a number of instances. Figure 2 of OpManager monitoring depicts that minimum CPU0 utilization was 02:02:48am, 02:47:48am, 03:02:48am, 05:17:48am, 05:47:48am, 06:02:48am, 07:02:48am, 07:17:48am, 08:02:48am, 08:32:48am, 09:02:48am, 09:17:48am, 11:32:48am, 11:47:48am, 01:02:48pm, 01:32:48pm, 01:47:48pm, 02:17:48pm, 02:32:48pm, 02:49:48pm, 03:17:48pm, 04:32:48pm, 04:47:48pm, 05:02:48pm, 06:02:48pm, 07:17:48pm, 07:32:48pm, 08:17:48pm, 08:32:48pm, 09:02:48pm, 09:32:48pm, 09:47:48pm, 10:32:48pm, 11:02:48pm, and 11:17:48pm. The maximum CPU0 utilization was measured and monitored at 10:02:48pm being 68.0%.

Similarly, in the figure below it is presented that the minimum amount of CPU1 utilization was 0.0%, maximum being 63.0% and the average was 3.87% within time interval of 00:00am to 22:00pm as mentioned above. Furthermore, we can learn from figure 2 that the minimum CPU1 utilization was measured and monitored by OpManager at a number of instances, similarly to CPU0. OpManager depicts that minimum CPU1 utilization was at 00:02:48am, 00:47:48am, 01:32:48am, 01:47:48am, 03:02:48am, 05:02:48am, 05:47:48am, 06:17:48am, 07:32:48am, 08:17:48am, 08:32:48am, 09:02:48am, 10:02:48am, 11:02:48am, 11:17:48am, 11:47:48am, 12:02:48pm, 12:32:48pm, 12:47:48pm, 01:02:48pm, 02:17:48pm, 03:17:48pm, 03:32:48pm, 03:47:48pm, 04:02:48pm, 04:17:48pm, 04:32:48pm, 05:32:48pm, 05:47:48pm, 06:32:48pm, 06:47:48pm, 07:47:48pm, 08:02:48pm, 08:17:48pm, 09:02:48pm, 09:17:48pm, 09:32:48pm, 10:32:48pm, 10:47:48pm, 11:02:48pm, 11:17:48pm, and 11:47:48pm. It is important to note that during the time interval of OpManager, the Walk-in Lab server was busy running and processing several requests from the lab users i.e. students, as this server runs Active Directory.

The figure below is a congruent graphical representation of CPU0 and CPU1 utilization as measured and monitored by ManageEngine OpManager.

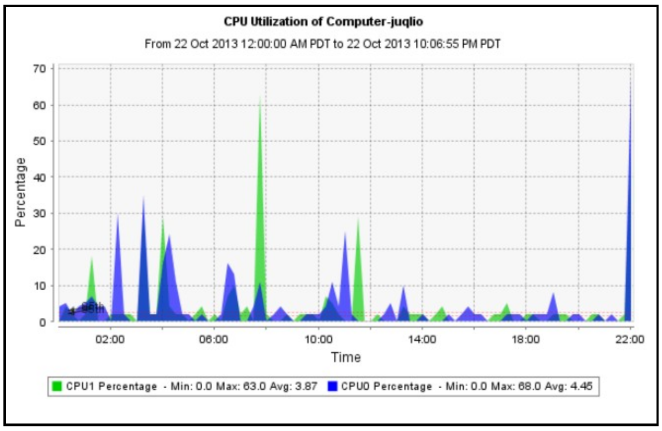

Figure 2: ManageEngine OpManager monitoring CPU utilization

Having made an account on the OpManager monitoring process, we now present the statistical monitoring results of Application Manager. Figure 3, below, depicts the measurement and monitoring of the CPU utilization by Application Manager and further presents the statistical results of such a process. The y-axis of the graph represents the amount of CPU utilization in percentages whereas the x-axis represents the time interval at

IJCSI International Journal of Computer Science Issues, Vol. 11, Issue 1, No 2, January 2014
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org

which the CPU utilization was measured and monitored by Application Manager. The time interval of the measuring and monitoring of the CPU utilization by Application Manager was ranging from 00:00am to 22:00pm whilst the percentage sufficed to range from 0% to 10%. In the figure below it is presented that the minimum amount of CPU0 utilization was 0.0%, maximum being 55.0% and the average was 3.687% within the aforementioned time interval. Furthermore, figure 3 presents that the minimum CPU utilization was measured and monitored by Application Manager once at 07:00pm being 0.0% and the maximum once being 9% at 03:00am.

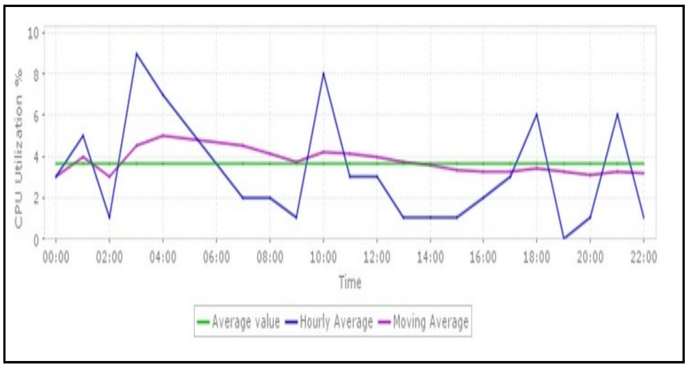

Figure 3: ManageEngine Application Manager monitoring CPU utilization

With regards to figure 4 of Heroix Longitude to follow, we present a graphical representation of the CPU utilization within a time interval of 04:00am to 10:00pm. The y-axis of the graph represents the amount of CPU utilization in percentages whereas the x-axis represents the time interval at which the CPU utilization was measured and monitored by Heroix Longitude. Heroix Longitude categorizes the CPU utilization into system CPU time and user CPU time usage hence why we hereby present the Heroix Longitude CPU utilization according to these two categories. For the purpose of clarity in providing a detailed account; we referred to the CPU System and CPU User as CPUsys and CPUuser respectively. Therefore, we have CPUsys utilization and CPUuser utilization to account on as part of the server components of Walk-in Lab server.

Figure 4 below depicts that the minimum amount of CPUsys utilization was 1.30% and the maximum being 3.40% within the time interval of 04:00am to 10:00pm as mentioned above. Heroix Longitude measured and monitored three instances of minimum CPUsys utilization of 1:30% at 04:48am, 05:53am and 06:03am accordingly. The maximum CPUsys utilization was measured and monitored to be 3.40% at 06:23am by Heroix Longitude.

Similarly to CPUsys utilization, in figure 4 the minimum amount of CPUuser utilization was 0.60% and the maximum being 4:40% within the time interval of 04:00am to 10:00pm as mentioned above. Heroix Longitude measured and monitored a single instance of minimum CPUuser utilization of 0:60% at 06:03am. The maximum CPUuser utilization was measured and monitored to be 4:40% at 06:23am by Heroix Longitude. The figure below is a graphical representation of the above mentioned results account.

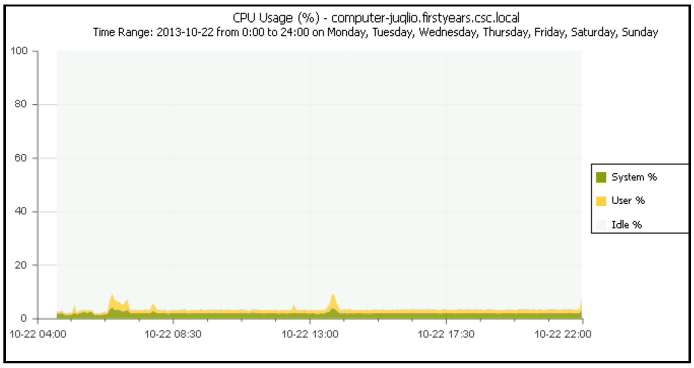

Figure 4: Heroix Longitude monitoring CPU utilization

1.2) Comparative Analysis: CPU utilization

The time interval of these three performance monitoring tools differs slightly on their start-time but their end-time is not different. ManageEngine OpManager and Application Manager started monitoring exactly at 00:00, while Heroix Longitude started monitoring about four hours later, at 04:00 to be precise, but both ended at 10:00pm. However this factor has no bearing or whatsoever on the data collected between the matching intervals i.e. from 04:00 to 10:00pm of these three performance monitoring tools. This, therefore, compelled us to consider the performance of these performance monitoring tools only on the matching time intervals as mentioned above under section [1.1] of CPU utilization results and thereby draw our analysis.

Within the 04:00am to 10:00pm time interval; OpManager measured and monitored 35 instances of minimum CPU0 utilization of 0.0% and 41 instances of minimum CPU1 utilization of 0.0%, Application Manager measured and monitored only one instance of minimum CPU utilization whereas Heroix Longitude measured and monitored 3 instances of minimum CPUsys utilization of 4.40% and only one instance of minimum CPUuser utilization of 0.80%.

Furthermore, the time at which these instances of minimum CPU utilization were monitored entirely do not correspond or whatsoever. For instance; the first instances, within the matching time interval, of minimum CPU utilization that OpManager measured and monitored were 05:17:48am and 05:02:48am for CPU0 and CPU1 respectively. Contrary, Application Manager measured and monitored the minimum CPU utilization at 07:00pm and for Heroix Longitude it was 06:03am and 06:03am for CPUsys and CPUuser respectively.

Additionally, the amount of minimum CPU utilization that OpManager, Application Manager and Heroix Longitude measured and monitored totally differs. OpManager, for both

IJCSI International Journal of Computer Science Issues, Vol. 11, Issue 1, No 2, January 2014
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org

processors, measured a minimum CPU utilization of 0.0% similarly to the minimum CPU utilization amount of 0% measured and monitored by Application Manager whereas Heroix Longitude measured a minimum CPU utilization amount of 1.30% and 0.60% for CPUsys and CPUuser respectively. The same analysis can be drawn for the maximum CPU utilization amounts that they monitored. OpManager measured a maximum CPU utilization amount of 68.0% and 63.0% for CPU0 and CPU1 respectively, for Application Manager the maximum CPU utilization amount is 55% whereas Heroix Longitude measured a maximum CPU utilization amount of 3.40% and 4.40% for CPUsys and CPUuser respectively.

2) *Memory Utilization*

In this section we present the results of the ManageEngine OpManager, ManageEngine Application Manager and Heroix Longitude monitoring with respect to the Memory utilization of the Walk-in Lab server. We base our presentation and comparative analysis on the information presented by figures 5, 6 and 7 to follow. We begin by making an account of the memory utilization with accord to OpManager, subsequently follow by Application Manager and finally make an account on the Heroix Longitude memory utilization.

1) *Results: Memory Utilization*

In figure 5 below, the y-axis of the graph represents the amount of memory utilization in percentages whereas the x-axis represents the time interval at which the memory utilization was measured and monitored by OpManager. OpManager measured and monitored the memory utilization from 00:00am till 10:00pm as we can see from figure 5. We can also learn from the very same figure that during this time interval, the minimum amount of memory utilization was 71.0%, maximum being 99.0% and the average of the memory utilization within this time interval was 88.0%. Furthermore, we can also learn that the minimum memory utilization was measured and monitored by OpManager at 05:47:46am whereas the maximum memory utilization was 05:32:46am. Figure 5 presents the aforementioned facts about OpManager performance monitoring in depth with the following shaded graph.

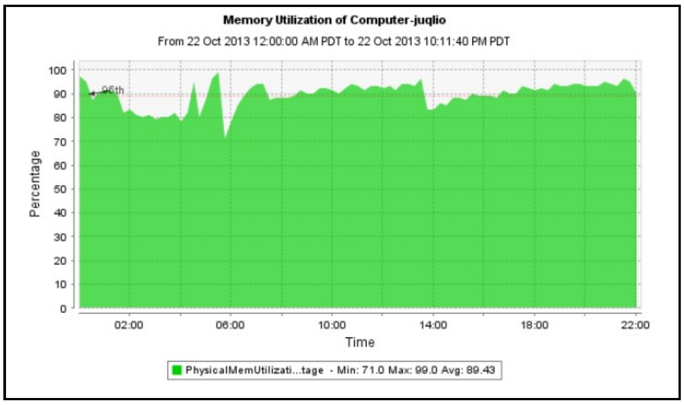

Figure 5 ManageEngine OpManager monitoring Memory utilization

Similarly, in figure 6 below, the y-axis of the graph represents the amount of memory utilization in percentages whereas the x-axis represents the time interval at which the memory utilization was measured and monitored by Application Manager. The time interval of Application Managers' memory utilization monitoring process started at 00:00am till 10:00pm as well. Figure 6 also depicts that during this time interval, the minimum amount of memory utilization was 72.0%, maximum being 97.0% and the average of the memory utilization within this time interval was 89.582%. Furthermore, the minimum memory utilization was measured and monitored by Application Manager at 07:00am whereas the maximum memory utilization was 02:00pm. Figure 6 below presents the discussed information about Application Manager's memory utilization.

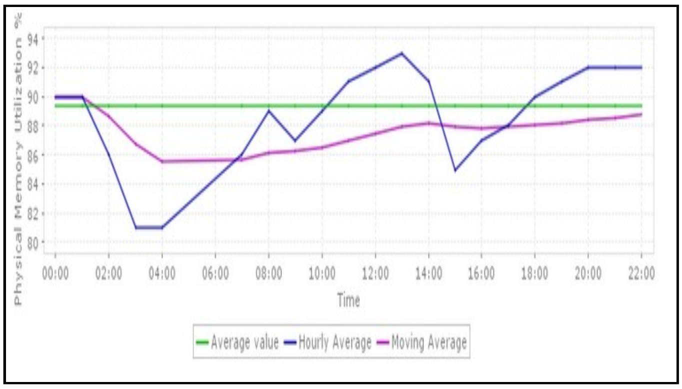

Figure 6: ManageEngine Application Manager monitoring Memory utilization

The following illustration in figure 7 presents a graphical representation of the memory utilization within a time interval ranging between 04:00am to 22:00pm as measured and monitored by Heroix Longitude. The y-axis of the graph represents the amount of memory utilization in megabytes whereas the x-axis represents the time interval at which the memory utilization was measured and monitored by Heroix Longitude. From the figure, we learned that the maximum memory utilization was measured and monitored at 05:43am being 2,001.00 MB and the minimum being 1,473.70 MB at 05:48am. Figure 7 presents a detailed account of the discussed memory utilization by Heroix Longitude.

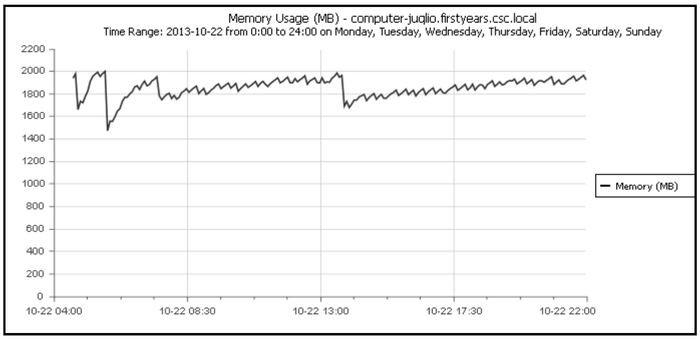

Figure 7: Heroix Longitude monitoring Memory utilization

2) *Comparative Analysis: Memory Utilization*

On the basis of the results presented above under section [2.1] of the results of CPU Utilization, it seems as if these three performance monitoring tools measured and monitored the memory utilization of two different servers. What informs this impression is that the statistical results they respectively present do not, at any point, show any similarity whatsoever but instead contradict each other.

To substantiate the claim articulated, OpManager measured and monitored the minimum amount of memory utilization as 71.0% at 05:47:46am and maximum being 94.0% at 05:32:46am. Application Manager measured and monitored the minimum amount of memory utilization as 72.0% at 07:00am and maximum being 97.0% at 02:00pm. Heroix Longitude measured and monitored the maximum memory utilization to be 2,001.00 MB at 05:43am and the minimum to be 1,473.70 MB at 05:48am.

Let us assume, for argument sake, that 99.0% or 97% by OpManager and Application Manager respectively represents 2.001.00 MB and 71.0% and 72% by OpManager and Application Manager respectively represents 1.473.70 MB of Heroix Longitude. We make this assumption on the basis that we want to acknowledge the fact that these three performance monitoring tools at least measured and monitored change in memory utilization. Then, perhaps a question can be posed here to say, does the measured and monitored change in the memory utilization bear any significance when measured and monitored at inaccurate instances? Is it enough to only measure and monitor such change in memory utilization inconsistently?

These questions can be posed on the basis of the statistical results presented by OpManager, Application Manager and Heroix Longitude. We noted with great surprise that the graph representations of these two performance monitoring tools about the memory utilization of the Walk-in Lab server, to a greater extent, varied in almost every moment of their monitoring. This inconsistency invoked questions about the accuracy of these performance monitoring tools and really condensed our level of reliability on them.

1) *Disk Utilization*

In this section we present the results of the ManageEngine OpManager, ManageEngine Application Manager and Heroix Longitude monitoring with respect to the Disk utilization of the Walk-in Lab server. Accordingly, we present the following three figures mentioned in this instance, namely Figures 8, 9 and 10, which depicts the graphical presentation of the disk utilization of the Walk-in Lab server as monitored by these three performance monitoring tools. We further take a journey of exploration scrutiny and analyzing these figures as we present them.

2) *Results: Disk Utilization*

We begin by figure 8 which depicts the OpManager disk utilization monitoring process. The y-axis of the graph represents the amount of disk utilization in percentages whereas the x-axis represents the time interval at which the disk utilization was measured and monitored by OpManager. The time interval of this figure ranges from 00:00am to 22:00pm and the disk utilization amount is represented in percentages ranging from 0% to 100%. According to OpManager the disk utilization of the Walk-in Lab server was constant between 00:00am and 10:00pm at the average of 22.00 %. There was no sudden change in the disk utilization or whatsoever according to this performance monitoring tool. Figure 8 below depicts this information:

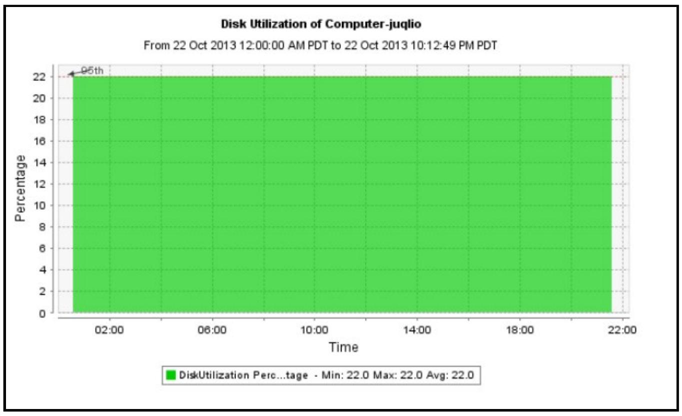

Figure 8: ManageEngine OpManager monitoring Disk utilization

Secondly, we present figure 9 which depicts the ManageEngine Application Manager Disk utilization monitoring process. Similarly to OpManager, the y-axis of the graph represents the amount of disk utilization in percentages whereas the x-axis represents the time interval at which the memory utilization was measured and monitored by Application Manager. However, the time interval of this figure ranges from 00:00am to 10:00pm and the disk utilization amount is represented in percentages ranging from 0% to 100%. According to ManageEngine Application Manager, the disk utilization of the Walk-in Lab server was constant between 00:00am and 10:00pm at the average of 22.00 %. This performance monitoring tool measured and monitored a sudden change in the disk utilization at 03:00am which lasted until 08:00am. However, this sudden change in the disk utilization failed to influence the average disk utilization as it remained constant throughout the time interval. The following illustration in Figure 9, presents this information in detail:

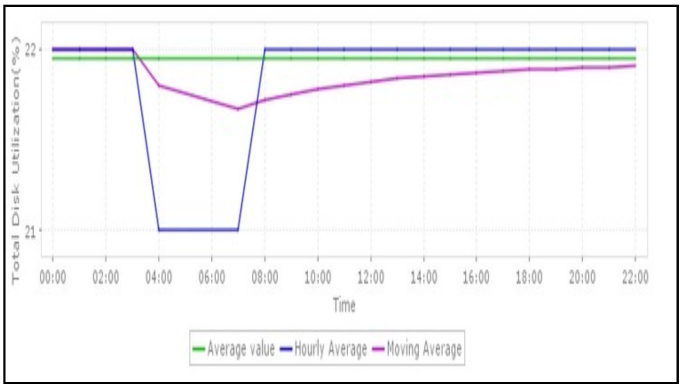

Figure 9: ManageEngine Application Manager monitoring Disk utilization

Finally, we present figure 10 which is about the disk utilization monitoring by Heroix Longitude. The y-axis of the graph represents the amount of disk utilization in megabytes whereas the x-axis represents the time interval at which the disk utilization was measured and monitored by OpManager. In figure 10 below, Heroix Longitude depicts the time interval of monitoring as ranging from 06:04am till 22:00pm and the average disk utilization as constant at 22, 539.00 MB. Figure 10 below depicts the discussed information about Heroix Longitude disk utilization monitoring.

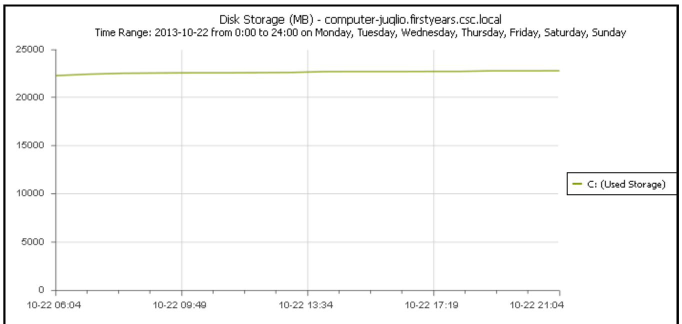

Figure 10: Heroix Longitude monitoring Disk utilization

1) *Comparative Analysis: Disk Utilization*

First and foremost, it is noteworthy to point it out here that it is next to impossible that, the disk utilization amount of the Walk-in lab server can consists of various versions between a certain given time interval. We therefore contend that if there exists only one version of the disk utilization amount, therefore there must only be one version of disk utilization statistics that are reported about such disk storage. The expectations about the performance monitoring tools that are monitoring one and the same computer components is that they will present one and the same statistical results of the components' performance, because the component performed itself in one accurate way.

Similarly, ManageEngine OpManager, ManageEngine Application Manager and Heroix Longitude, on average, measured and monitored indifferently because the statistical outcomes are similar. However, OpManager monitoring might be misleading to network administrators to some extent because, drawing from figure 8, OpManager only presents results on average leaving an impression that nothing strange is taking place in the disk utilization behind the scenes.

VIII. CONCLUSION AND FUTURE WORK

In this section we make a conclusion on the basis of the above three scrutiny and comparative analysis conducted under the section of performance scrutiny. The comparative analysis was conducted on the basis of the three performance monitoring tools, namely, ManageEngine Application Manager, OpManager and Heroix Longitude. Drawing from the comparative analysis conducted under this performance study, there have existed some serious levels of inconsistencies between the ManageEngine products and Heroix Longitude. The levels of inconsistencies were extreme as we can still learn about this from chapter seven of the performance study above. These levels of inconsistencies invoke serious questions about a number of critical aspects underpinning these performance monitoring tools. They pose questioning on the accuracy, reliability, quality assurance and to a greater extent, their legitimacy.

It is said that "Accurate performance monitoring and reporting is critical in a virtual infrastructure, where many VMs compete for limited host resources" [12]. This is a general principle underpinning the performance monitoring tools. However, their lack of accuracy, not on the part of collecting the necessary data from system resources but, on reporting the collection of such data as it happened can jeopardize the business life of an organization. "Understanding the performance monitoring tools that they supposedly help to identify and analyze faults and performance issues" [13], it becomes very difficult to rely on them when they are inaccurate.

Finally, we conclude by making the following expert guidance:

- It is for the good of the business organizations to compare the statistical results provided by the performance monitoring tools with the server baseline.

- It is vital for organizations and Network Administrators to spend more time trying to determine the accuracy level of the performance monitoring tools prior relying on them.
- Business organizations should adopt monitoring redundancy though deploying at least two performance monitoring tools into their servers. This sounds costly for organizations but this approach ensures better monitoring because it provides a broader insight on what is occurring behind the scenes.
- With the redundancy monitoring, the intervention of the Network Administrators on server and application performance can either be informed by performance inconsistencies and/or consistencies. What do we mean by this? By this we wean, a Network Administrator can either make an intervention whenever each performance monitoring tool detects an anomaly or else intervene when all of them detect an anomaly.

From the comparative analysis conducted under the performance study in chapter seven of this paper; we have seen consistency and similar statistical results amongst the ManageEngine produced performance monitoring tools. However, their consistency runs parallel to non-ManageEngine produced performance monitoring tool and this leaves a lot to be desired.

On the basis of our conclusion, we present the following as the future work emerging from this research work. We hold a conviction that there is still a room for further research under the research field of performance monitoring tools and as a result we hereby present the following critical issues as cases for future work:

- What are the technical factors impacting the performance of the performance monitoring tools such that they present different statistical results about one and the system resources?
- Does the time interval for monitoring bear any amount of influence on the statistical results and their presentations?
- The possibilities of inventing a new Agentless server and application performance monitoring tool that will accurately predict the future performance of a server using the predicting coding.
- Measure the extent of being Agentless of the Agentless server and application performance monitoring tools

IX. ACKNOWLDGMENTS

We wish to extend our utmost words of gratitude to the University of Fort Hare, Computer Science Department for having granted us with an opportunity to undertake this research project on their departmental facilities.

X. REFERENCES

[1]. Keshav T, "A Survey of Network Performance Monitoring Tools." (2006) Available on:http://www.cse.wustl.edu/~jain/cse567-06/ftp/net_perf_monitors/index.html

[2]. Debbie Russo. "Basic Application and Server Monitoring Federal Center Consolidation Initiative (FCCI)" Available:http://solarwindsarketing.s3.amazonaws.com/solarwinds/whitepapers/FDCCI Whitepaper .pdf

[3]. CICS Transaction Gateway for z/OS Version 8 Release 0 Information Center
Available:http://pic.dhe.ibm.com/infocenter/cicstgzo/v8r0/index.jsp?topic=%2Fcom.ibm.cics.tg.zos.doc%2Fctgzos%2Fccl99ch.html

[4]. Seliverstov, D., John T. and Tianxiang Z. "The Pros and Cons of Collecting Performance Data using Agentless technology." *Int. CMG Conference*. 2007.

[5]. Anderson, T., Peterson, L., Shenker, S., & Turner, J. Overcoming the Internet impasse through virtualization. *Computer*, *38*(4), 34-41. (2005).

[6]. Menascé, D. A. Virtualization: Concepts, applications, and performance modeling. In *Int. CMG Conference* (pp. 407-414). (2005).

[7]. Fraenkel, J. R., Wallen, N. E., & Hyun, H. H. "How to design and evaluate research in education." (1993).

[8].Okanović, D. et al. "SLA-driven adaptive monitoring of distributed applications for performance problem localization." *Computer Science and Information Systems* 10.1: 25-50. (2013)

[9]. Ammons, G., Ball, T., Larus, J. R.: Exploiting Hardware Performance Counters With Flow and Context Sensitive Profiling. In Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI '97). ACM, Las Vegas, Nevada, USA. 85-96. (1997)

[10].Pape, C., & Trommer, R. Monitoring VMware-based Virtual Infrastructures with OpenNMS. (2012).

[11]. CPU, Memory and Disk Monitoring
Available on: http://www.manageengine.com/network-monitoring/cpu-memory-disk.html

[12]. Koster, K.J. (2013). Types of Monitoring
Available on: http://www.kjkoster.org/Blog/Types_of_Monitoring.html

[13]. Siebert, E. Why physical performance monitoring tools aren't enough. (2010).
Available on: http://searchservervirtualization.techtarget.com/tip/Why-physical-performance-monitoring-tools-arent-enough

Phumzile Nomnga originates from Engcobo, Eastern Cape Province, South Africa. He received his Bachelor of Science degree in 2012 from the University of Fort Hare and is currently studying towards his Bachelor of Science (Honours) in Computer Science degree at the same institution.

Mfundo Shakes Scott is the Head of the Department of Computer Science and a Researcher of the Telkom Centre of Excellence at the University of Fort Hare. His research interests are network management systems, ICT4D, and e-Service provisioning.