3.1 Introduction

It is very important to keep the software services available for customers with the highest possible performance. The system is able to detect the issues automatically so that it is possible to handle for customer to use the system with performance satisfaction. To monitor the network and hardware devices with software services are more crucial tasks to manage the IT infrastructure.

Large Companies and enterprises are hugely dependent on the IT industries for providing the profit to them, often they make a dedicated monitor center. In this manner it is possible to respond quickly when issue have been raised – and resolve them before becoming real problems, regardless of the complexity and size of system being monitored, which requires few important programs or systems to complete this job. In this chapter, we will focus to find the problems and different aspects for right monitor software with Linux guests under the z/VM on the mainframe [62][64].

3.2 Components used for monitoring Mainframe Systems

Error conditions/ Standard warning

It displays typical conditions that give the results in errors or warning on the normal servers, e.g. UNIX servers and Microsoft Windows servers. This is not taking duration, severity of event account, number of repetition, although like as thresholds/levels typically can be specified.

Processor

The usage of Processor can be monitored to detect the general resource shortages and system irregularities. Moreover, concrete error condition includes:

- CPU usage with High system, e.g. above 80%
• Low idle time

Memory

It is ultimately an important resource. If there is a problem, then system will run out of memory. It is a negative impact on the system performance if system will need to swap constantly, because working active set makes bigger than physical memory. Examples of error conditions:

• System will be low then make free swap space.
• The page rate will be to/from swap is higher.

File system, Disk and files

Disks can be monitored to ensure the space for persistent along with temporary data. In few situations, monitor includes physical strongest of disk devices to ensure data integrity. In this, unexpected periods can come with higher amounts of writes/reads may show I/O bottleneck or to fail the application which is out of control. Some individual files can monitor for changes to take the security breaches and these files are important for system operations. Concrete examples of the warning in category:

• Disk low on the available space
• Disk low on i/o nodes
• For too many reads or write need High disk utilization
• Modification of files

Network Connectivity and interfaces

A general well-function server has not enough, if services are running on it but cannot be reached, due to the network makes malfunction. Examples, which raises errors:

• High Packet loss.
• Network interface down.
• Many packet checksum errors or packet collisions

_Processes, programs services and logs_

The previous category is focused on the individual programs. If main process, program or service does not found the process list which is expected, this is indication of error. In some situations it can be useful to filter the log files to find the error conditions. Sometimes, it is important to call the real service and ensure that it gives the results within adequate response time.

• Unexpected high processor utilization of programs/daemons
• High sets of zombie processes
• Special entries occur in log files
• Service will not respond as expected in acceptable time.

3.3 Mainframe/virtualization considerations

In virtual environment running servers such as z/VM need some consideration of what to monitor, where to do it, and how to do it [64].

_Processor_

The usage of processors readings is an example of monitoring the data that cannot be delivered within Linux guest. Therefore Linux kernels will never include “steal time” which is definitely not with the old kernels. In the virtual environments, it is dependent upon old kernels that are important to trustworthy usage of CPU reading from CP layer.

_Hypervisor data_

Monitor from individual guests that will not be able to take useful information about virtual environment like as whole. In this, data from hypervisor level provides trustworthy picture of virtual machines together [80].
Crucial values can include the I/O rates, CP page activity, and CPU utilization. Moreover, z/VM can provide common method to access the CP monitor data for their programs such as “IBM Performance Toolkit” and many more monitoring products [78].

**Instrumentation overhead**

The focus on resource sharing and consumption naturally increases when running virtual servers, and it becomes quite evident, if the monitoring software consumes processor cycles and memory away. If instrumentation will be overhead for the single Linux instance coming near to the processor, the monitor system will quickly consume whole processors depending upon number of guests.

The license costs of remaining software portfolio increases with the addition of some extra processors, and eventually monitor software will become expensive. Monitoring solutions should be fast, small and efficient and need to have preferably small memory of footprint.

**3.4 Support, security, manageability, costs**

There are various problems that need to be taken into account while selecting the monitoring software: Once for Product support. For any company, it is crucial to have a dependable support unit that helps to resolve problems, answer questions reasonably fast and offers fixes to the impending software bugs.

It is a problem for open source software, unless software can be included in enterprise class Linux support and distribution. With the noticeable increase in number of monitored system, uncomplicated management becomes another important factor. The task of logging in on potentially hundreds of servers rapidly becomes time-consuming and gravelly monotonous, while updating the monitoring software or making small configuration adjustments.

Hence the importance of Central Management in the system being monitored becomes an important feature, if not already included. Monitoring software are likely to pose security threat. For instance, it is important to slacken off the firewall rules to allow transfer of monitoring data or manage the monitoring software itself. Therefore, high security will become a vital part of monitoring solution. Finally costs will be evaluated – and not from
the purchase or software license. Customization, Implementation and management costs will be assessed as well.

This chapter touches on the topic of performance of mainframe and their services – focus the monitoring software according to the company wishes. As the monitoring concept will be considered as an offset for investigation, it was not possible to perform detailed study of topic in due time.

All goals of thesis are achieved except when it comes to search monitoring software solution. This thesis provides the information in detail and considerations of initial monitoring that will be useful to fulfill their goals.

- Provide a concise, practically and theoretically balanced, system virtualization description on mainframe using z/VM.
- Formulate relative method to estimate distribution of the processor resources b/w individual virtual machines in the z/VM.
- Supply the number of developed tools and test programs.
- A robust, little, multithreaded, and memory consuming on-demand, test program that will estimate to the memory access rate and helps to take Linux swap with z/VM page performance issues.
- A customizable solution to monitoring practical of internal Linux values – with use of Linux components.
- A monitor client program that can run the log and plot monitored data.
- Provide tangible test result for generally accepted optimization methods, while giving some insight in run z/VM and Linux.

3.5 Summary

During the literature review we have observed various attributes which toil as overhead and proposed new possibilities, to regard the old problems. We will also showcase the performance aspects of these attributes with different propositions and compared the
results with the performance tools available in market like IBM Performance Tool and Bonnie Benchmark Tool [78][79].