International Journal of Engineering Research and Development
e-ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com
Volume 10, Issue 11 (November 2014), PP.49-53

Analysis of IT Monitoring Using Open Source Software Techniques: A Review

Ravikant sahu[1], Samta Gajhbhiye[2]
[1]*M.E. student, Computer Science and Engineering Department, CSVTU, Bhilai FET, SSGI, Bhilai, India*
[2]*Prof., Computer Science and Engineering Department, CSVTU, Bhilai FET, SSGI, Bhilai, India*.

Abstract:- The Network administrators usually rely on generic and built-in monitoring tools for network security. Ideally, the network infrastructure is supposed to have carefully designed strategies to scale up monitoring tools and techniques as the network grows, over time. Without this, there can be network performance challenges, downtimes due to failures, and most importantly, penetration attacks. These can lead to monetary losses as well as loss of reputation. Thus, there is a need for best practices to monitor network infrastructure in an agile manner. Network security monitoring involves collecting network packet data, segregating it among all the 7 OSI layers, and applying intelligent algorithms to get answers to security-related questions. The purpose is to know in real-time what is happening on the network at a detailed level, and strengthen security by hardening the processes, devices, appliances, software policies, etc. The Multi Router Traffic Grapher, or just simply MRTG, is free software for monitoring and measuring the traffic load on network links. It allows the user to see traffic load on a network over time in graphical form.

Keywords:- network security, packet, monitoring, segregating, network.

I. INTRODUCTION

The present deals with the methodologies and approaches adopted by the researchers in the field of IT monitoring system using OSS light weight MRTG., It has been observed that the classification processes of network and server cluster have been done using graphical approaches. In the recent time it has been also observed that soft-computing technique OSS is making room in the classification process of IT monitoring systems. Most researchers have adopted either enterprise level or licenses based. In this area of research, a very little amount of work has been done using open source software coding technique. So for this one technique have been done for IT industry by most of the researchers.

II. LITERATURE SURVEY

Several works on passive measurement systems have been proposed in the literature over the last few years. The IT monitoring of network and server cluster has great significant to explore types of plant and explain the usage of open source software for very new ideas share in best performance and optimization for this techniques witch in a major concern of today's and future's mankind, however it is very time consuming task, which is usually executed by present botanists. For monitoring hues network and server cluster in any lager IT originations relevant discriminating features. Most time best services perform a proactive services a client side. The best light weight and very fast performance low cost and very reliable this research work is focus.

Although the task of monitoring whether a device is up or down may seem relatively simple, the potential consequences of the outcome of this check can be huge for an organization reliant on IT systems. The intention of this article was to demonstrate how IT monitoring functions from a tiered approach, from simple monitoring of whether a device is responsive, through to using Proactive services 24x7 to view the health of services, view cost of a service being down, and to see how business service availability may impact an organization.

From the literature it has been also found that IT monitoring graphical light weight representing [1] M. Kodialam, T.V. Lakshman, Dynamic routing of restorable bandwidth-guaranteed tunnels using aggregated network resource usage information, IEEE/ACM Transactions on Networking 11 (3) (2003) 399–410. [2] M. Jain, C. Dovrolis, End-to-end available bandwidth: measurement methodology, dynamics and relation with TCP throughput, in: Proceedings of ACM SIGCOMM, August 2002. [3] M. Jain, C. Dovrolis, Pathload: a measurement tool for end-to-end available bandwidth, in: Proceedings of Passive and Active Measurement Workshop, March 2002. [4] N. Hu, P. Steenkiste, Evaluation and characterization of available bandwidth probing techniques, IEEE JSAC Special Issue in Internet and WWW Measurement, Mapping, and Modeling 21 (6) (2003) 879–894. [5] C. Dovrolis, P. Ramanathan, D. Moore, What do packet dispersion techniques measure? Proceedings of IEEE INFOCOM 2 (Apr.) (2001) 905–914. [6] V. Ribeiro, R. Riedi, R. Baraniuk, J. Navratil, L. Cottrell, pathChirp: efficient available bandwidth estimation for network paths, in: Proceedings of Passive and

Active Measurements Workshop, April 2003. [7] J. Strauss, D. Katabi, F. Kaashoek, A measurement study of available bandwidthestimation tools, in: Proceedingsof Internet Measurement Conference, Miami, Florida, October 2003. [8] T. Oetiker, Monitoring your IT gear: the MRTG story, IEEE IT Professional 3 (6) (2001) 44–48. [9] G. Jin, B.L. Tierney, System capability effects on algorithms for network bandwidth measurement, in: Proceedings of Internet Measurement Conference, Miami, Florida, October 2003. [10] M. Jain, C. Dovrolis, Ten fallacies and pitfalls on end-toend available bandwidth estimation, in: Proceedings of Internet Measurement Conference, Taormina, Italy, and October 2004. [11] L. Angrisani, S. D鋒Antonio, M. Vadursi, G. Ventre, Performance comparison of different techniques for available bandwidth measurement in packet-switch ed networks, in: Proceedings of VECIMS鋒03, Lugano, Switzerland, July 2003, pp. 212–217.

III. PROBLEM IDENTIFICATION AND SOLUTION METHODOLOGY

All IT monitoring using open source software's techniques is very difficult task when different.

This situation prompted the development of the Multi Router Traffic Grapher. Every five minutes, it queried the ''Octet Counters'' of the university's internet Gateway router. From this data, the average transfer rate of the Internet link was derived for every five minute interval and a web page was generated with four graphs showing the transfer rates for the last day, week, month, and year. The visual presentation on the

Web allowed everyone with a web browser to monitor the status of the link. presents an MRTG-generated web page. While the availability of these graphs did of course not increase the capacity of the link, the performance data provided by MRTG proved to be a key argument to convince management that a faster Internet link was indeed needed.

A. How MRTG works

The MRTG logged its data to an ASCII file, rewriting it every five minutes, constantly consolidating it, so that the log file would not grow over time. The log file did only store slightly more data than was needed to draw the graphs on the web page. The graphs ere Converted to GIF format by piping a graph in PNM format to the pnmto gif tool from the PBM package. This setup limited MRTG to monitor about 20 router ports from a workstation.

A second obstacle for potential users was that MRTG required snmp get from the CMU NMP package. This package proved to be rather difficult to compile on various platforms at that time. In the meantime, I had left De Montfort University and was working at the Swiss Federal Institute of Technology. There I had no responsibility for the campus network and the Internet link was sufficiently fast. MRTG was not one of my top priority projects anymore. Because the CMU SNMP library did not compile on Solaris I had not even a working installation of MRTG. This all changed when Dave Rand <daver@bungi.com> got interested in MRTG and contributed a small C program called rate up. Rate up solved MRTG's performance problem by implementing the two most CPU intensive tasks in C and thus moving them out of the MRTG Perl script. Rateup did the logfile rewriting and the graph generation. Rate up initiated the development of MRTG-2.x. First, I modified rate up to use Thomas Boutell's GD library [5] which enabled it to generate GIF files much faster than pnmtogif. Second, the SNMP Portability problem was solved by switching from CMU' ssnmpget to Simon Leinen's Perl SNMP module [4], written in pure Perl and thus making it virtually platform independent.

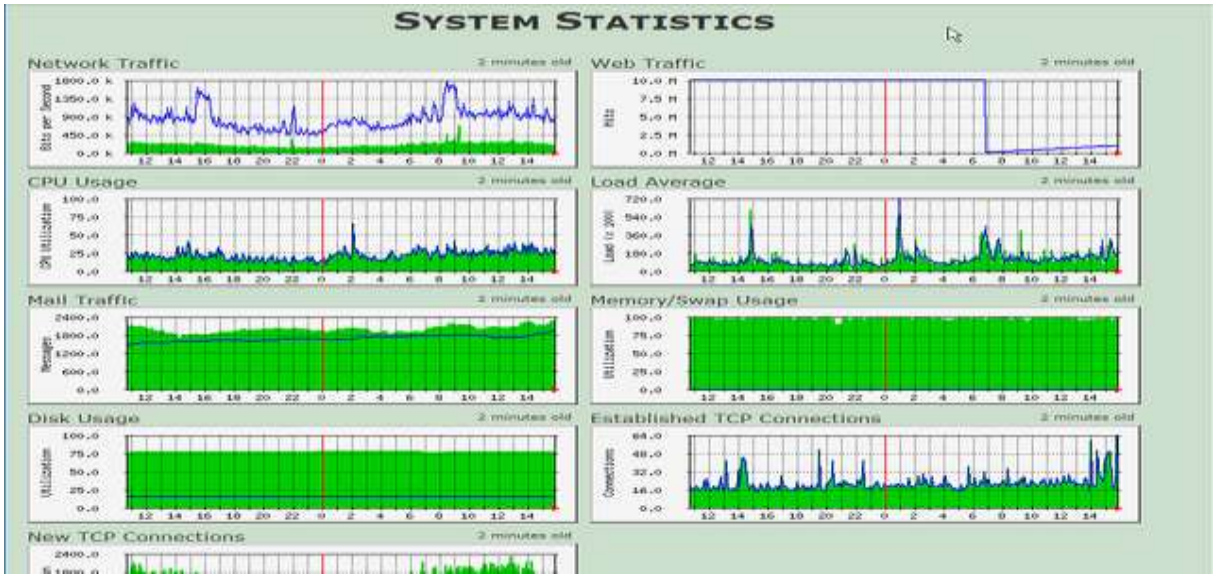

Fig 1 Multi Router Grapher (MRTG) Graphical Presentation

B. The IT monitoring system working module

IT monitoring allows you to create your own views to group the devices logically and manage them from one place. This option helps you manage the devices under each geographical location and assign authorized access to the business views. A Network delay in preventing a fault from occurring, or repairing a damage in a lesser turn-around time requires a fool-proof alerting mechanism where the concerned engineer gets to know the source of the problem by way of a meaningful alert. Scalability of the Solution All said and done, a network monitoring solution must not take a beating and crash or it must do so with a warning at the least!A server on which you host the monitoring solution, or the monitoring application itself is as susceptible as the other resources on the network. Having a redundant server take over and provide un-interrupted monitoring.

IV. IMPLEMENTATION RESULTS AND DISCUSSIONS

The experimental setup of the work that has been carried over NIC Chhattisgarh state Data centers IT monitoring system. The developed open source code are based on the near most efficient 70% other IT monitoring software's and very optimal that a compare.

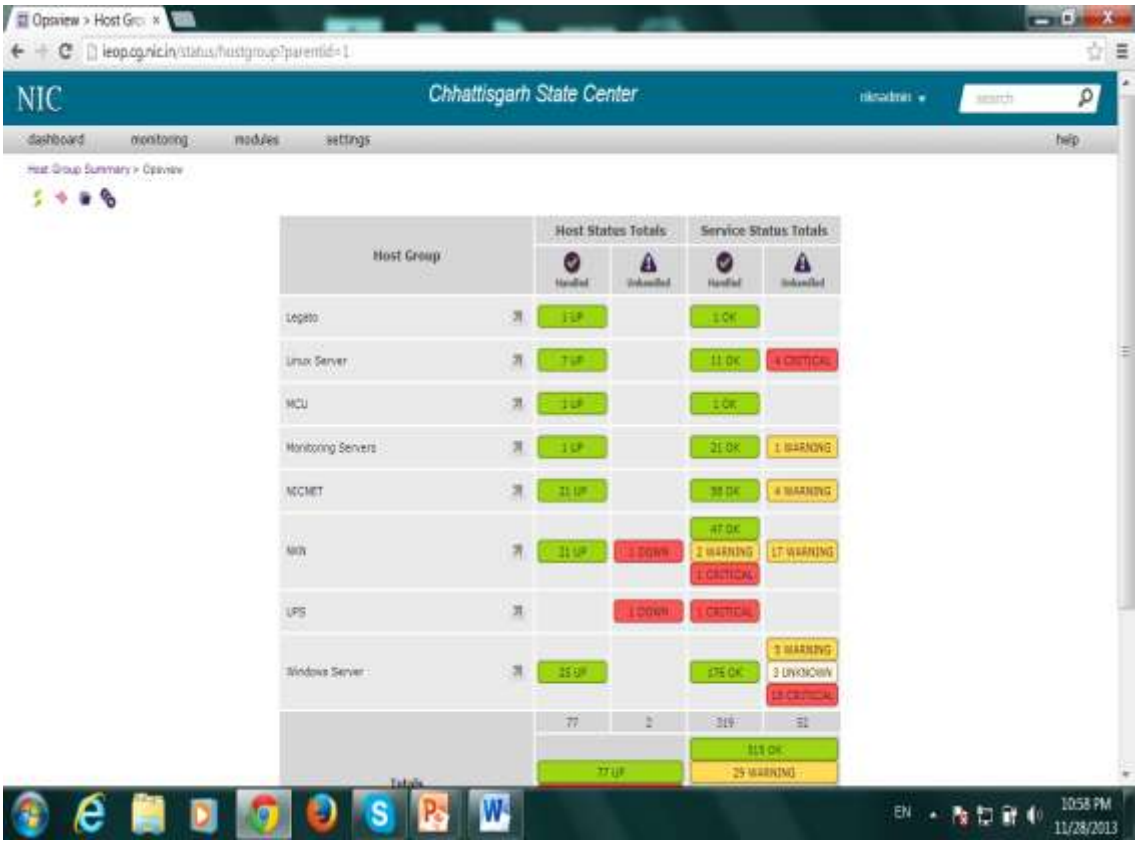

Fig. 2 Host status server and network cluster

So now we have templates to reduce our time-to-value, our only time consuming task is adding these hosts "hostname by hostname". Again, innovation in monitoring has deemed this an unnecessary evil – with the creation of auto discovery.

Auto-discovery, in its most simplistic form, allows a monitoring system to "go out" and scan a predefined subnet or network and find devices on that network. In our Windows example, we can scan the subnet and discover all **hosts** on that network and import them into our monitoring system, ready to be modified and have host templates added to.

The development of IT monitoring system knowledge-base consists of various steps. The experimental setups involved in framing of knowledge-based module.

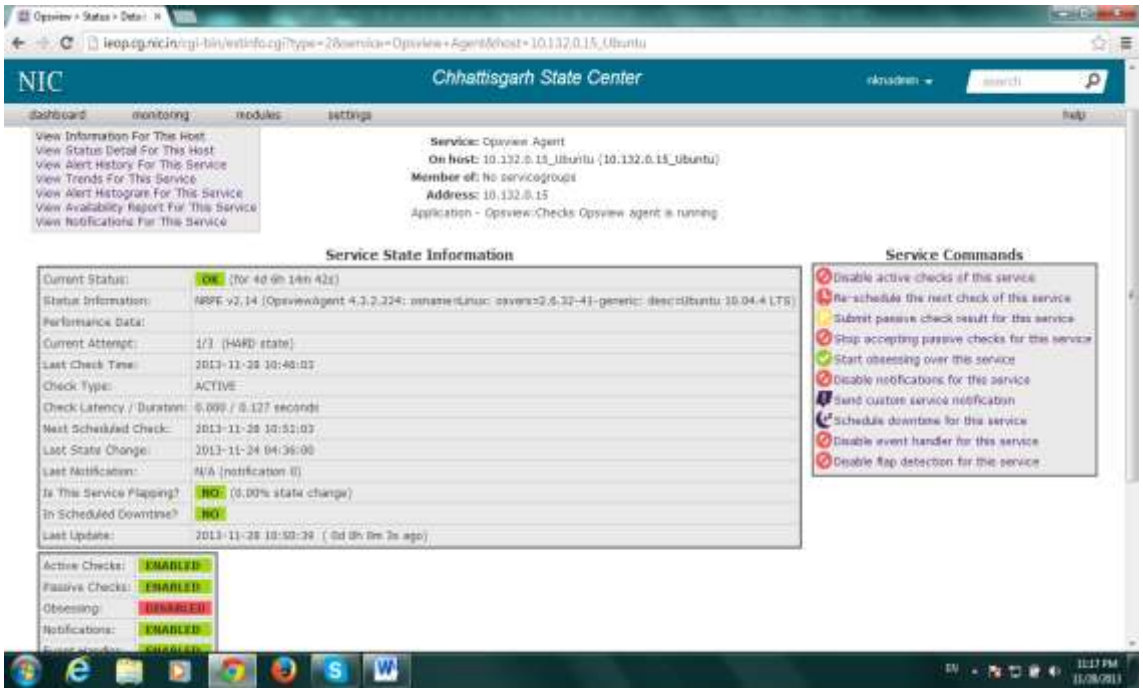

Fig 3 Server cluster service status

This gives us a great view into how each of these components is doing. What we want now is to look at it holistically; for example as a website rather than a series of objects. To do this, Monitoring software vendors / APM vendors have created "business process monitoring".

V. CONCLUSION AND FUTURE SCOPE

Considerable amount of work has been done IT monitoring system using different approaches. The present work has carried out for IT monitoring of open source software's techniques base on Nagios Core: Provides the core set of monitoring and alerting capabilities and used monitoring engine. Perl: The primary programming language used Catalyst: Web application framework used for building the web application MySQL: A relational database used for configuration, runtime (Berkley Light Wight Database) Net-SNMP: Provides SNMP support MRTG : Provides lightweight 2D graphing Nagvis Network infrastructure Graph Webmin Web base Desboard Controlling utility OS : Linux.

For further study, IT monitoring network cluster and server cluster open source software's community allays new updates forums and modification the new concept of IT monitoring system best optimum solution the 24x7 monitoring for IT service providers proactive services that a client location best and fast new ideas. The development open source code has to be very flexible. It means it not only work with present research problem but beneficial for other research problems.

ACKNOWLEDGMENT

The real spirit of achieving a goal is through the way of excellence and austerious discipline. I want to thank SSGI, Bhilai, India, for providing me the necessary software, tools and other resources to deliver my research work. I acknowledge with gratitude and humanity my indebtedness to **Prof. Samta Gajbhiye,** FET, SSGI, SSTC, Bhilai, under whose guidance I had the privilege to complete this paper..

REFERENCES

[1]. A. Danalis and Constantinos Dovrolis. Anemos: An autonomous network monitoring system. In In Proceedings of the 4th Passive and Active Measurements (PAM) Workshop, April2003.

[2]. Aditya Akella, Srinivasan Seshan, and Anees Shaikh. An empirical evaluation of wide-areaInternet bottlenecks. In Proc.ACM IMC, October2003.

[3]. Kostas G. Anagnostakis, Michael B.Greenwald, and RaphaelS.Ryger. cing: Measuring network-internal delays using only existing infrastructure. In Proc. IEEE INFOCOM, April2003.

[4]. GiuseppeDi Battista, Maurizio Patrignani, and Maurizio Pizzonia. Computing the types of the relationships between autonomously stems. InProc. IEEEINFOCOM, April2003.

[5]. Jean-ChrysostomeBolot.End-to-endpakcetdepayandlossbehaviorintheInternet.InProc.ACM SIGCOMM'93, San Francisco, CA, September1993.

[6]. Tian Bu, Nick Duffield, Francesco Lo Presti, and Don Towsley. Network tomography on general topologies. In Proceedings of the 2002 ACM SIGMETRICS international conference on Measurement and modeling of computer systems, pages 21–30.ACM Press, 2002.

[7]. R.Carterand M.Crovella. Measuring bottleneck link speed in packet-switch works. Technical report, Boston University Computer Science Department, March 1996

[8]. Meeyoung Cha, Sue Moon, Chong-Dae Park, and Aman Shaikh. Positioning Relay Nodesin ISP Networks. In Proc. IEEE INFOCOM, March 2005.

[9]. Y. Chen, D. Bindel, H. Song, and R. H. Katz. An algebraic approach to practical And scalable over laynet work monitoring. In Proc. ACM GCOMM,August2004.

[10]. Yanghua Chu, Aditya Ganjam, T. S. Eugene Ng, Sanjay G. Rao, Kunwadee Sripanid kulchai, Jibin Zhan, andHui Zhang. Early experience withan Internet broadcast system based on overlay multicast. In Prof. of USENIX Annual Technical Conference, June2004.

[11]. Leonard Ciavattone, Alfred Morton, and Gomathi Ramachandran. Standardized Active measurements on a tier 1 ip backbone. In IEEE Communications Magazine, pages90–97, San Francisco, CA, June2003.