JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Survey on Remote On-Premise Server Monitoring

Favaz Mohammed Rawoof,
School of CS & IT, Jain University,
Bangalore, Karnataka, India

Mohd Tajammul,
School of CS & IT, Jain University,
Bangalore, Karnataka, India

Abstract: *Server Monitoring is monitoring resource metrics like CPU usage, Memory usage, Disk Space, Disk Health, Network Bandwidth and Connected Devices of On-Premise Servers or Cloud Servers. This paper surveys on-premise servers which lack remote monitoring to identify and mitigate any outages or downtime that would affect business continuity. On-premise servers require security as it contains sensitive information. This requires it to be kept in a separate room with supervision and security checks, making it difficult to monitor it periodically to catch any issues before it disrupts business continuity.*

By making monitoring data available remotely via a Telegram bot, monitoring the servers becomes easy on-site or off-site and only requires the administrators to physically inspect the server for initial configuration, periodic checks and any physical repairs. Monitoring software like Prometheus can be used to collect server metrics and send them to an authorized user in Telegram. Telegram provides an API for bots that can send notifications to its user. The server administrator can be notified periodically on the status of the server or if the status reaches a critical metric threshold set by the administrator I.e., notifies the administrator when the server CPU temperature reaches 95°C.

Index Terms – On-Premise Server, Remote Monitoring, Telegram Bot, Python Service, Prometheus.

I. INTRODUCTION

Telegram is a cross-platform, open-source, cloud-based messaging service launched in 2013. The platform provides end-to-end encrypted messaging, voice and video calls, and an API to automate replies to users [8]. Server monitoring metrics can be formatted in a mobile-friendly format and sent as notifications. This eliminates the need to be physically in the server room for monitoring the status of on-premise servers.

Prometheus is a monitoring and alerting open-source toolkit [9]. This software allows to collect monitoring data and store it in a database, this data can be further pushed to other services to analyze, view or share it remotely.

Python services can do various tasks like fetching data from databases and servers, interacting with Telegram bots to send and receive messages, etc. [10].

The aim is to create a python service that can be set up to send notifications regarding monitoring data, alerts and other data extracted from Prometheus when requested by an authorized user on Telegram or when the metrics cross a certain threshold set by the server administrator. This enables the System Administrator to monitor servers remotely and be up to date on the status of the server anytime, anywhere.

II. RELATED WORKS

The paper “Monitoring System of OpenStack Cloud Platform Based on Prometheus” has a similar view on monitoring servers that are on the open stack cloud. The paper designs an efficient way to monitor the open stack cloud platform by using tools like Prometheus and visualize it by using Grafana [3]

The paper “Network’s Server Monitoring and Analysis Using Nagios” states the importance of monitoring network topologies and server metrics at regular intervals using NAGIOS, avoiding any system failure by notifying the network administrator via e-mail.[2]

A research “Implementation System Telegram Bot for Monitoring Linux Server”, uses a Telegram bot to execute direct terminal commands in a Linux server environment allowing the author to remotely monitor the server. This method requires its users to memorize the commands beforehand and then send it to the bot. [1]

In a research paper titled “Design of Telegram Bots for Campus Information Sharing”, the author created a telegram bot for a university to mass circulate upcoming events, classes, timetable and other information among staffs, teachers and students easily. The author shows how easy it is to share information in a technologically advanced era where most of the users have an online messaging app like Telegram.[4]

A paper titled “Remote Monitoring and Control by Embedded Database Design and Web Server Implementation Using SQL Lite database and boa webserver”, ports a boa webserver to an arm Linux platform and uses SQL lite to remotely monitor virtually anything with appropriate sensors by using a portable pocket-sized development board MINI2440. [5]

Remote monitoring is crucial in areas other than servers, a paper titled “Design and Implementation of Remote Monitoring System for Welding Machines based on Web”, The authors tackle the problem of monitoring clusters of welding machines in their workshop. By creating a web server using HTML, JavaScript, PHP, Ajax and connecting them to welding machine endpoints, the system helps to monitor welding stations at mass and mitigate any issues.[6]

In a survey paper titled “Network and Server Resource Management Strategies for Data Centre Infrastructures” the author dives deep through various resource management methods data centers choose to make deploying, monitoring and running virtual machines efficiently. The paper shows various methods of monitoring virtual machines in datacenters.[7]

Authors have discussed various aspects of implementing cloud security and best practices to be followed on cloud platforms like Amazon Web Services and Microsoft Azure, aiming to protect user information. [11-24]

III. PROBLEM STATEMENT

Traditional On-Premise Servers lack Remote Online Monitoring Dashboards Like Amazon CloudWatch and Azure Insights. Monitoring on-Premise servers are crucial as modern-day on-premise servers are deployed to store and access confidential and sensitive data that can’t be made public. This sensitive data is accessed on-premise and is required to be available most of the time for Business Continuity. Thus, the server requires it to be monitored periodically to avoid any suspicious activity or downtime.

IV. PROPOSED SYSTEM

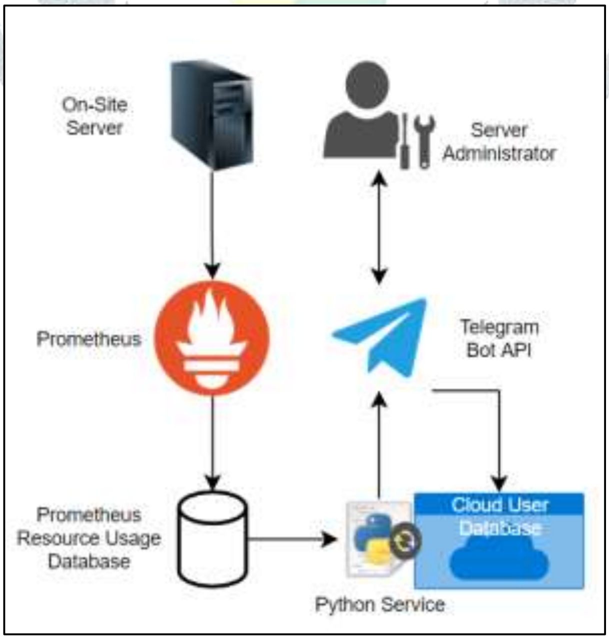

Figure 1: System Architecture

The system architecture as shown in ***Figure 1*** uses open-source monitoring software in this case Prometheus, that collects a wide range of system resource data and stores it in a database.

A python service is used to check a database in the cloud if a registered Telegram user is paired with the server key created by the service. The pairing process can be initiated when starting the bot in Telegram App. Since Telegram is a widely used open-source messaging platform, it discards the need to install any additional app for monitoring. When the python service finds a Telegram user paired with the system key in the cloud database, it formats and sends the extracted data from the Prometheus database to the paired user.

Notifications can be sent as per the interval or thresholds set in the python service by the system administrators. Thus, allowing remote monitoring of an on-premise server.

V. CONCLUSION

In this paper, we surveyed various methods of monitoring an on-premise server remotely. This enables us to understand the impact of lack of server monitoring which might lead to a wide range of issues like Overloading, Overheating, Data loss, Malicious attack, etc. in absence of the server administrator.

Various authors have mentioned methods for monitoring servers and other IOT devices by installing monitoring software like Prometheus, NAGIOS etc. The recorded data can be further used to measure resources usages or investigate any suspicious incidents that might occur.

We have also looked through various methods authors have proposed to make this local monitoring data and other information available online using dedicated web servers and social platforms like Telegram, enabling remote monitoring.

The monitoring data can be further processed by a python service that pushes the formatted data to the registered Telegram user by a Telegram bot.

The proposed solution provides system administrators with an easy and secure way to monitor on-premise servers remotely.

VI. FUTURE SCOPE

Additional features that can be added to the proposed solution in the future is as follows:

- Introduce compatibility with other monitoring tools.
- Introduce compatibility with other messaging platforms.
- Notify multiple messaging platforms and administrators simultaneously.

REFERENCES

[1] Mohammad Idhom, Ronggo Alit, Henni Endah Wahanani, Akhmad Fauzi "Implementation System Telegram Bot for Monitoring Linux Server", 2018

[2] J. Renita, N. Edna Elizabeth, "Network's Server Monitoring and Analysis Using Nagios", 2017

[3] Lei Chen, Ming Xian, Jian Liu, "Monitoring System of OpenStack Cloud Platform Based on Prometheus", 2020

[4] Hari Setiaji, Irving V Paputungan, "Design of Telegram Bots for Campus Information Sharing", 2018

[5] Roopa Bammidi, Deepa Kundala, "Remote Monitoring and Control by Embedded Database Design and Web Server Implementation Using SQL Lite database and boa web server", 2017

[6] Zhang Yu-Long, Cao Heng, Hu Jing-Feng, Wang Jin-Cheng, "Design and implementation of remote monitoring system for welding machine based on web", 2018

[7] Fung Po Tsoa, Simon Jouetb, Dimitrios P. Pezaros, "Network and Server Resource Management Strategies for Data Centre Infrastructures: A Survey", 2017

[8] Telegram, https://core.telegram.org/bots, 2021

[9] Prometheus, https://prometheus.io/docs/introduction, 2021

[10] CoreTechnologies, https://www.coretechnologies.com/products/AlwaysUp/Apps/RunPythonScriptAsAService.html, 2021

[11] Alam T., Tajammul M., Gupta R. (2022) Towards the Sustainable Development of Smart Cities Through Cloud Computing. In: Piuri V., Shaw R.N., Ghosh A., Islam R. (eds) AI and IoT for Smart City Applications. Studies in Computational Intelligence, vol 1002.

[12] Tajammul, M., Shaw R.N., Ghosh A., Parveen R. (2021) Error Detection Algorithm for Cloud Outsourced Big Data. In: Bansal J.C., Fung L.C.C., Simic M., Ghosh A. (eds) Advances in Applications of Data-Driven Computing. Advances in Intelligent Systems and Computing, vol 1319.

[13] Tajammul, M, Parveen, R., "Cloud Storage in Context of Amazon Web Services", International Journal of All Research Education and Scientific Methods, vol. 10, issue 01, pp. 442-446, 2021.

[14] Tajammul, M., Parveen, R., "Auto Encryption Algorithm for Uploading Data on Cloud Storage", BIJIT - BVICAM's International Journal of Information Technology, vol. 12, Issue 3, pp. 831-837, 2020.

[15] Tajammul, M., Parveen, R., "Key Generation Algorithm Coupled with DES for Securing Cloud Storage," International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249-8958, Volume-8 Issue-5, June 2019 no. 5, pp. 1452–1458, 2019.

[16] Tajammul M., Parveen R., "Two Pass Multidimensional Key Generation and Encryption Algorithm for Data Storage Security in Cloud Computing", International Journal of Recent Technology in Engineering, Vol. 8, Issue-2, pp. 4152–4158, 2019.

[17] Tajammul M., Parveen R., "Algorithm for Document Integrity Testing Pre-Upload and Post- Download from Cloud Storage", International Journal of Recent Technology in Engineering, Vol. 8, Issue-2S6, pp. 973–979, 2019.

[18] Tajammul, M., Parveen, R., "Auto Encryption Algorithm for Uploading Data on Cloud Storage", BIJIT - BVICAM's International Journal of Information Technology, vol. 12, Issue 3, pp. 831-837, 2020.

[19] Tajammul, M., Parveen, R., and M. Shahnawaz, "Cloud Computing Security Issues and Methods to Resolve: Review," Journal of Basic Applied Engineering and Research, vol. 5, no. 7, pp. 545–550, 2018.

[20] Tajammul, M., Parveen, R., Delhi, N. (2018). Comparative Study of Big Ten Information Security Management System Standards, International Journal of Engineering Research in Computer Science and Engineering (IJERCSE) Vol 5, Issue 2, pp. 5-14, 2018.

[21] M. Tajammul, R. Parveen, N. K. Gaur and S. D, "Data Sensitive Algorithm Integrated with Compression Technique for Secured and Efficient Utilization of Cloud Storage," 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), 2021, pp. 1-9, doi: 10.1109/GUCON50781.2021.9573648.

[22] Tajammul, M., Parveen, R., (2017). Comparative Analysis of Big Ten ISMS Standards and Their Effect on Cloud Computing, 978-1-5386-0627 8/17/31:00c2017IEEE; 9001; 362367.

[23] Tajammul, M., and R. Parveen, "To Carve out Private Cloud with Total Functionality," 2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), 2020, pp. 831-835, doi: 10.1109/ICACCCN51052.2020.9362826.

[24] M. Tajammul, R. Parveen and I. A. Tayubi, "Comparative Analysis of Security Algorithms used in Cloud Computing," 2021 8th International Conference on Computing for Sustainable Global Development (INDIACom), 2021, pp. 875-880, doi: 10.1109/INDIACom51348.2021.00157.