International Journal of Advanced Scientific Innovation Volume 05 Issue 01, FEB 2023
ISSN: 2582-8436

# Private cloud with live Performance MonitoringAnalysis

Pranav T P, Venkatesh Prasad

*Department of Computer Science & Engineering, PES University*

*Banglore, Karnataka, India.*

Pranavparamesh010@gmail.com

***Abstract - The open-source monitoring program Prometheus is available. Open-Stack is becoming the most popular open source cloud platform because to ongoing developments in cloud computing. In order to guarantee the reliability and stability of cloud platforms when utilised as production systems, thorough system monitoring is an essential link and a critical method. Along with the capabilities of the Open-Stack cloud platform, this article uses Prometheus to collect monitoring data, takes use of the real-time monitoring data visualization application grafana, and develops a comprehensive, clever, and efficient monitoring system. Testing is a viable method for improving the dependability and stability of the Open-Stack cloud platform.***

**Keywords — Prometheus, Open-Stack, Cloud Platform, Grafana.**

## I. INTRODUCTION

Cloud computing is a new business computing model. Cloud computing platforms share vast infrastructure, data storage, and software, creating vast pools of resources from which users can obtain computing power, storage space, and information services on demand. Based on this, hierarchical services are abstracted to provide users with services such as infrastructure (IaaS), platform (PaaS) and software (SaaS) in the form of payment. Cloud services should be easy and convenient to use, as people don't have to worry about the specific implementation of the underlying system, they just need to put a lot of processing power and memory into "cloud" processing, and they return processing results. guaranteed. These days, companies like Google, Amazon, and IBM are aggressively pushing their own cloud services. Cloud computing has evolved from concept to real-world application implementations into personalized, service-oriented public or private clouds.

Monitoring is an important part of cloud computing platforms and a prerequisite for network analysis, system management, job scheduling, load balancing, event prediction, failure detection, and recovery operations on cloud computing platforms. Monitoring helps cloud computing platforms dynamically quantify resource usage, detect service flaws, identify user usage patterns, and resource planning which plays an important role in improving service quality of cloud computing platform.

Most existing monitoring techniques are designed only for monitoring grid clusters, which generally have the characteristics of single data, poor real-time performance, and low efficiency. Compared with the characteristics of cloud computing platforms, it cannot meet the needs of practical applications. A comprehensive, intelligent and efficient monitoring system not only brings a lot of convenience to the operation and maintenance of the cloud platform, but also an important guarantee of the stability and reliability of the cloud platform.

## II. OPEN-STACK

### A. Open-Stack Introduction

Open-Stack originally consisted of Nova, developed by NASA, and Swift, developed by Rack space and later licensed by Apache. It's an open code project for building public and private clouds and managing cloud platforms. The Open-Stack community, which includes server makers, Linux distributors, network equipment makers, storage makers, and other IT giants, is more active than its competitors, and new projects continue to evolve in the form of incubation, making it the most popular and widely used open source cloud platform.

### B. Open-Stack Architecture

Open-Stack isolates users across the network from the vast hardware resources behind the network. On the one hand, Open-Stack is responsible for interacting with the hypervisor running on the physical nodes to provide management and control over various hardware resources, and on the other hand, providing users with virtual machines that meet their needs. To do. So far, Open-Stack has 10 components integrated. The cloud can be managed with a web-based dashboard, which allow administrators to control, provision, and automate Open-Stack resources. These components include the compute service Nova, the

network service Neutron, the object storage service Swift, the block storage service Cinder, the image service Glance, the authentication service Keystone, the UI service Ceilometer, the cluster service Heat. The relationship between components is shown below in Figure 1.

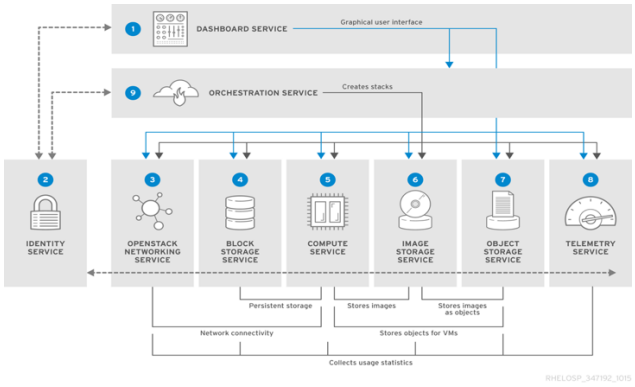


Fig. 1. Open-stack Architecture

## III. PROMETHEUS

### A. Prometheus Introduction

Prometheus is a collection of tools for monitoring and alerting open source systems. Prometheus was created by Sound Cloud and has been used by a number of companies and organizations since its birth in 2012. This open source project has a large number of active developers and community users. It is now a stand-alone open source monitoring project maintained by each company. In order to make the project more substantial and clear, it joined cloud native Computing Foundation in 2016 and became the second member of the organization after k8s.

### B. Prometheus core Architecture

Prometheus main modules include Prometheus Server, Exporter, Pushgateway, PromQL, Alertmanager and WebUI. Prometheus servers get data from statically configured targets or targets from server discovery. DNS, K8s, Mesos, etc. Exporters are responsible for scheduling data reports to Prometheus servers. Pushgateway can pull all target data into different subnet environments and aggregate the data collected by Prometheus in a unified way. Alertmanager implements Prometheus alert functionality and WebUI mainly implements data display. Prometheus core architecture is shown in Figure 2.

### C. Prometheus Constraints

1) ***High Quality Data :*** A high-dimensional data model is used by Prometheus. A set of key-value pairs and the name of the measure are used to identify a time series.

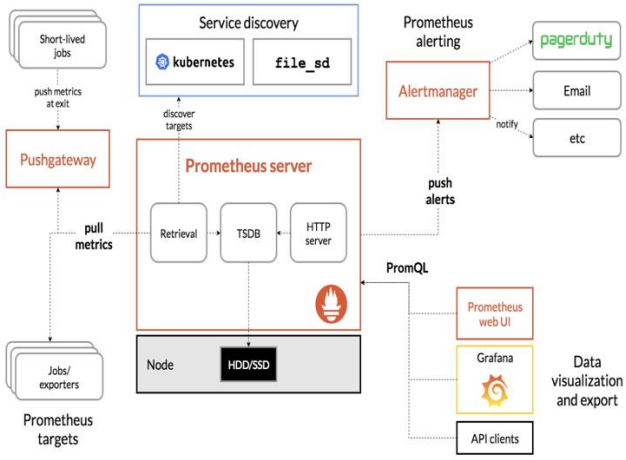


Fig. 2. Prometheus Core Architecture

2) ***Effective inquiries*** : Time series data may be sliced and diced using Prometheus' versatile query language to produce specialized graphs, tables, and alerts.

3) ***Exceptional Visualization*** : Prometheus provides a number of visual data patterns, including a console template language, an integrated expression browser, and Grafana connectivity.

4) ***Efficient Storage :*** Prometheus uses an effective proprietary format to store the time series in memory and on the local disc.

5) ***Accurate Alert :*** The alerts are created using the versatile query language of Prometheus and the upkeep of dimensional data. Notifications are managed by the alert administrator.

6) ***Numerous Client Libraries :*** The Client libraries allow easy use of services. Prometheus already supports more than ten languages, and the custom libraries are easy to implement.

### D. Grafana Constraints

Grafana is a cross platform open-source measurement analysis and visualization tool, which can query the collected data and display it visually, and notify in time. It has six characteristics.

1) ***Display Mode :*** The Quick and flexible client charts, panel plugins have many different ways of visualizing indicators and logs, the official library has a wealth of dashboard plugins, such as heat maps, line charts, charts and other display ways.

2) ***Data Source :*** Data sources include Graphite, InfluxDB, OpenTSDB, Prometheus, Elasticsearch, CloudWatch and KairosDB.

3) ***Notice Reminder :*** Grafana visually defines the alert rules for the most important indicators.

Grafana will continuously calculate and send notifications, and get notifications through Slack, PagerDuty, etc., when the data reaches the threshold.

4) ***Mixed Display :*** Grafana can mix different data sources in the same chart, and can specify or even customize data sources based on each query.
5) ***Annotation :*** Grafana is able to annotate charts from different data sources. Hovering over an event will display the complete event metadata and Tags.
6) ***Filter :*** Ad-hoc filters allow the dynamic creation of new key/value filters, which are automatically applied to all queries using the data source.

## IV. CONCLUSIONS

The above design implements a efficient monitoring system, intel- ligent and by utilizing the powerful monitoring tool Prometheus and the visualization tool grafana. The experiment proves that the system has good effectiveness and live performance, which makes the Open-Stack cloud platform more complete and also helps to maintain the stability and reliability of Open-Stack cloud platform**.**

## REFERENCES

*[1] Wang L, Von Laszewski G, Younge A, et al. Cloud computing: a perspective study[J].New Generation Computing, 2010, 28(2): 137-146.*
*[2] Han FF, Peng JJ, Zhang W, Li Q, Li JD, Jiang QL.Virtual resource monitoring in cloud computing. Journal of Shanghai University (Engl Ed), 2011, 15(5): 381385.*
*[3] Rossigneux F, Lefevre L, Gelas J P, et al. A generic and extensible framework for monitoring energy consumption of OpenStack clouds[C]//Big Data and Cloud Computing (BdCloud), 2014 IEEE Fourth International Conference on. IEEE, 2014: 696-702*
*[4] Prometheus, https://prometheus.io/docs/introduction/overview/, 2014-2020.*
*[5] Chris Madden Grafana, https://grafana.com/grafana/, 2020.*
*[6] Girish, L., Rao, S.K.N. Anomaly detection in cloud environment using artificial intelligence techniques. Computing (2021). https://doi.org/10.1007/s00607-021-00941-x*
*[7] FutureGrid, https://portal.futuresystems.org. W. Gentzsch,Sun Grid Engine: towards creating a compute power grid, 1st IEEE/ACM Int. pp. 35 36, May2001..*
*[8] C. David, Introduction to DevOps on AWS, AWS, pp. 2, Dec. 2014 https://d0.awsstatic.com/whitepapers/AWSDevOps.pdf*
*[9] Sentinel Cloud Service, Sentinel Cloud V3.5 Quick Start Guide, Sentinel, Jun2014*
*[10]Elastic Compute Cloud(EC2), Amazon Web Services, Amazon. com, http://aws.amazon.com/ec2.*