Tencent Managed Service for Prometheus

Product Introduction

Product Documentation

Contents

Product Introduction

Overview

Last updated：2021-12-16 14:55:15

Tencent Managed Service for Prometheus (TMP) provides the highly available Prometheus service as well as the open-source visualization tool Grafana and Cloud Monitor alarms while inheriting the monitoring capabilities of the open-source Prometheus, which reduce your development and OPS costs.

Prometheus Overview

Prometheus is an open-source monitoring system. Similar to Kubernetes inspired by Google's Borgmon monitoring system, it was inspired by Google's Borgmon monitoring system. It was created and developed in 2012 by SoundCloud's internal engineers and released in January 2015. In May 2016, it became the second project after Kubernetes to officially join Cloud Native Computing Foundation (CNCF). Nowadays, it is usually used for monitoring in the most common Kubernetes container management systems.

Prometheus has the following features:

- Custom multidimensional data models (a time series data entry is composed of a metric and a key-value pair called label).
- Flexible and powerful query language PromQL, which can use multidimensional data to complete complex monitoring queries.
- Independence from distributed storage and support for operations based on one single master node.
- Time series data collection through HTTP pull.
- Data push through Pushgateway.
- Acquisition of collection target servers through dynamic scrape configuration or static configuration.
- Integration with Grafana to easily support various visual charts and dashboards.

Features

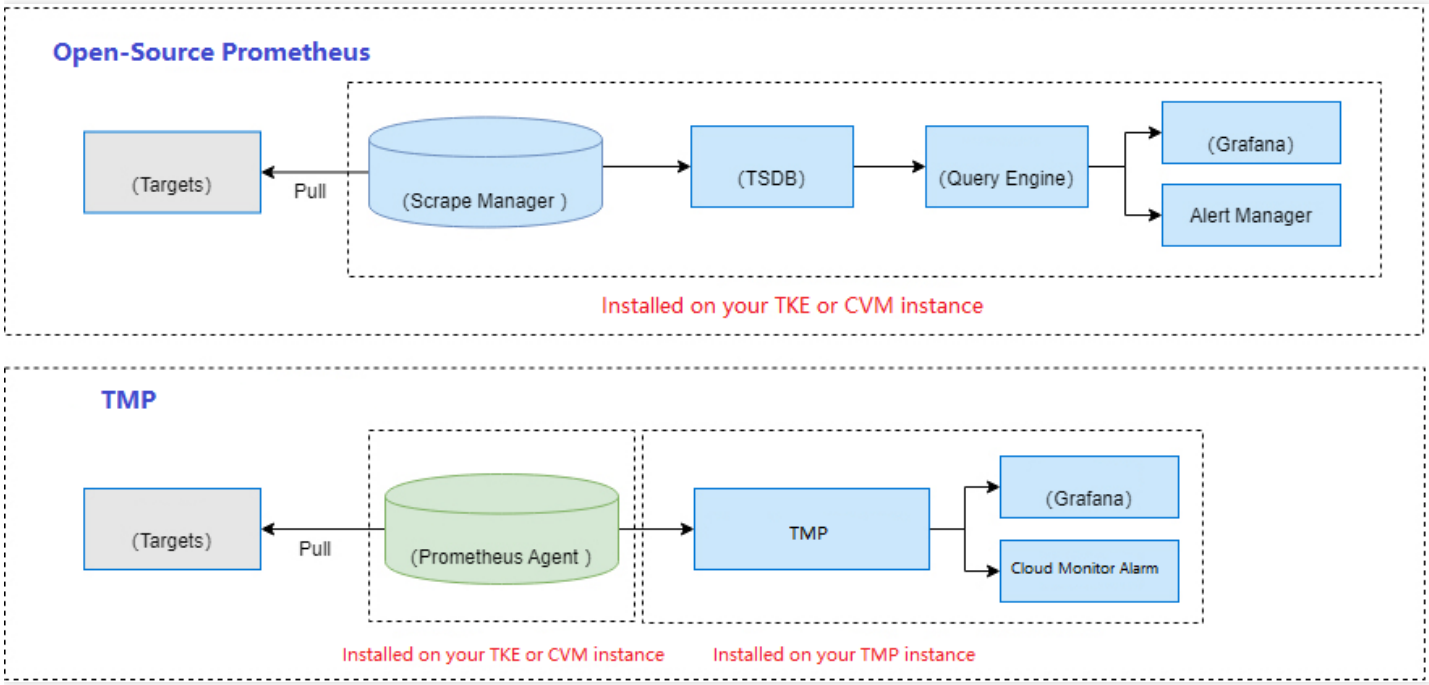
According to the layering of monitoring, TMP covers business monitoring, application layer monitoring, middleware monitoring, and system layer monitoring. Together with Cloud Monitor's alarming capabilities and open-source Grafana, it can provide a one-stop all-round monitoring system to help your quickly identify and locate business problems and reduce the impact of various faults on your business.

- System layer monitoring: CPU, memory, disk, network, etc.

- Intermediate component layer monitoring: Kafka, MySQL, Redis, etc.
- Application layer monitoring: application services such as JVM, HTTP, and RPC.
- Business monitoring: golden business metrics such as the number of logins and the number of orders.

Strengths

TMP has the following strengths over the open-source Prometheus:



- Lighter, more stable, and more available.
- Fully compatible with the open-source Prometheus ecosystem.
- Free of manual setup, saving the development costs.
- Highly integrated with TKE, saving the development costs of integrating with Kubernetes.
- Integrated with Cloud Monitor's alarming system, saving the costs of developing alarm notifications.
- Integrated with commonly used Grafana dashboards and alerting rule templates.

Strengths

Last updated：2021-12-16 14:55:15

Lightweight Service

Compared with the open-source Prometheus monitoring service, TMP features a more lightweight overall structure. You can use a TMP instance simply after creating it in Cloud Monitor, where an agent can complete data scrape by using less than 1 GB of memory.

High Stability and Reliability

TMP only occupies megabytes of resources, much fewer than the open-source Prometheus does. It also integrates Tencent Cloud storage services and its own replica capabilities to reduce the number of system interruptions and provide services with a higher availability.

Great Openness

TMP offers the out-of-the-box Grafana service. It also integrates a wealth of Kubernetes basic monitoring services and dashboards for common service monitoring, which can be quickly used after activation.

Low Costs

TMP provides the native Prometheus monitoring service. After you purchase a TMP instance, you can quickly integrate it with TKE to monitor services running on Kubernetes, which eliminates the costs of setup, OPS, and development.

High Scalability

TMP features an unlimited data storage capacity not restricted to local disks. It can be dynamically scaled based on Tencent Cloud's proprietary sharding and scheduling technologies to meet your elastic needs. It also supports CLB for better load balancing. This helps solve the pain points that the open-source Prometheus cannot be scaled horizontally.

High Compatibility

TMP is fully compatible with Prometheus protocols, support core APIs, custom multidimensional data models, flexible query language PromQL, and collection target discovery through dynamic scrape configuration or static configuration, so you can easily migrate to it for smooth access.

Use Cases

Last updated：2021-12-16 14:55:15

Integrated Monitoring

TMP provides the one-stop out-of-the-box Prometheus monitoring service, which is natively integrated with Grafana dashboards and Cloud Monitor alarms for various monitoring scenarios such as basic services, application layer, and container services.

Application Service Monitoring

Scenario 1

An application provides external API services but their quality information cannot be secured. TMP can be integrated according to the development language to monitor the access request volume, delay, and success rate of APIs in real time.

Scenario 2

TMP also detects service exceptions to help you understand which APIs an exception responds to, which servers an exception occurs on, and whether an exception occurs on individual servers or in the entire cluster.

Scenario 3

For Java applications, TMP can monitor the GC, memory, and thread status of individual servers to help you fully understand the internal status of JVM.

CVM Monitoring

If your service is deployed in CVM, you must modify the Prometheus scrape configuration almost every time the service is scaled. For this kind of scenario, with the aid of the tagging capability of the Tencent Cloud platform and the tag discovery capability of the Prometheus agent, you only need to properly associate tags with CVM instances to easily manage and monitor target objects, which helps you get rid of continuously updating the configuration manually; for example:

1. Service A is deployed on 2 CVM instances at the same time, and the instances are associated with the tag "service name: A";
2. The original number of CVM instances cannot meet the requirements of a business campaign, and 3 more instances should be added. At this time, you only need to associate the tag "service name: A" with these new instances, and then the agent will automatically discover them and actively scrape their monitoring metrics;
3. After the campaign is over, 3 CVM instances are removed, and the agent can automatically discover the instance deactivation and stop scraping their monitoring metrics.

Custom Monitoring

You can use TMP to customize the reported metric monitoring data so as to monitor internal status of applications or services, such as the number of processed requests and the number of orders. You can also monitor the processing duration of some core logic, such as requesting external services.

Limits

Last updated：2022-08-23 11:26:29

Instance limits

Each instance can have up to 4.5 million series. If you need to adjust it, submit a ticket for application.

> Note：
> A series consists of a metric name and label. The same metric name and labels form a unique series.

Custom reporting limits

If you use TMP's custom monitoring feature to monitor data, there will be the following limits on metrics (series with a unique `__name__`).

- Data reporting must carry a metric name, i.e., the `__name__` label, which can contain only ASCII letter, characters, digits, underscores, and colons and must start with a letter and match the regex `[a-zA-Z_:][a-zA-Z0-9_:]*`. For more information, see Metric names and labels.
- Each metric can have up to 32 labels.
- The label name can contain only ASCII letter, characters, digits, underscores, and colons and must match the regex `[a-Za-Z_][a-Za-Z0-9_]*`. Labels starting with `__` are for internal use only.
- The label name and label value can contain up to 1,024 and 2,048 characters respectively.
- Under the same metric, the dimension combinations of labels cannot exceed 100,000. When the histogram has many buckets, the histogram-type metrics cannot be adjusted.

> Note：
> The role of labels: In Prometheus, data is stored as time series, which are uniquely identified by the metric name and a series of labels (key-value pairs). Different labels represent different time series, so you can query the specified data by label. The more labels you add, the finer the query dimension.

Prometheus query limits

To ensure the query efficiency and better user experience, Prometheus query has the following limits (which don't apply to metadata such as queries about labels and don't affect the Grafana metrics browser feature).

- The number of time series involved in a single query cannot exceed 100,000.
- The amount of data involved in a single query cannot exceed 100 MB.
- There is no limit on the query frequency, but if the concurrency exceeds 15, there may be a certain queuing delay in slower large queries (the probability is low though). Large queries with a time span of more than two weeks will have a higher delay.

The above limits also apply to alarm rules and recording rules. We recommend you limit the query scope based on your business scenario or appropriately split large queries in other ways. You can also use the method of splitting first and then aggregating, such as aggregating recorded results again.

Other configuration limits

Configuration limits:

- Up to 150 alarm rules can be configured for each instance.
- Up to 150 recording rules can be configured for each instance.